詳細・明解
パソコンシリーズ

いっしょうけんめ

一生懸命
ファミリーコンピュータ™
ゲーム＋（プラス）ベーシック入門

木村香奈枝・斉藤 千秋 著

●本書をお読みの方へ

●本書は**ファミリーコンピュータ**をよりよく利用するための入門書です。特に**ファミリーベーシック**をお持ちで、BASICを理解したいと思われている方には最適です。

●カートリッジゲームの特選攻略法、正当的キーボード入門法『タッチメソッド』など、ファミコンを活用するうえでのすべてを解説してあります。

●本書の特長は「やさしく詳しく」説明してあることです。やさしく、詳しい。この一見反対のようなことを、本文では必要なことをやさしく、注釈文では本文では説明できなかった解説を詳しく説明。注釈文については技術スタッフが担当しています。

●このように「やさしく詳しい」本書は低年令の方から大人の方までじゅうぶんにご満足いただけることでしょう。

いっしょうけんめ

一生懸命

ファミリーコンピュータ™

ゲーム＋（プラス）ベーシック入門

詳細・明解
パソコンシリーズ

ファミコンゲームランド

ゲーム攻略法アルバム
＋
オリジナルゲームカット

バンゲリング・ベイ

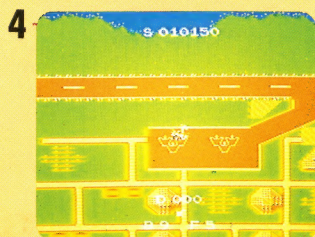
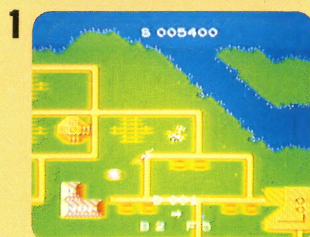
1

2

3

4

①
工場周辺のレーザー、高射砲をまず破壊します。遠くの工場から攻撃するのがベストです。

②
WORNING表示されたら、スグ、敵「戦艦」を発見し攻撃！戦艦に出航されたらアウト。

③
誘導ミサイルは、旋回してよけます。そのまま逃げると、追撃される可能性大です。

④
敵飛行場に着陸して、爆弾を補給することもできます。空母が遠い場合に利用します。

F-1レース

1

2

3

4

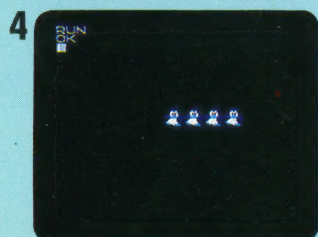
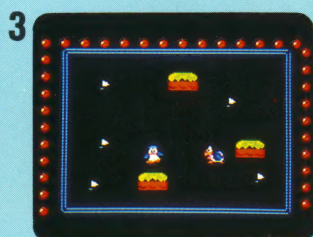
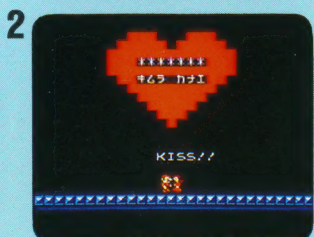
①
L字コーナーは、外側から入って、コーナー内側ギリギリをかすめるのがベストです。
②〜④
最上のスピードをたもつように、ブレーキとアクセルをたくみに操作。わずかでも短距離で走れるように、コーナーを曲がる。この2点が高得点テクニックです。例えば、ヘアピンコーナーでは、外側から入り内側へ、内側から外側へと抜けるようにします。

オリジナル

1

2

3

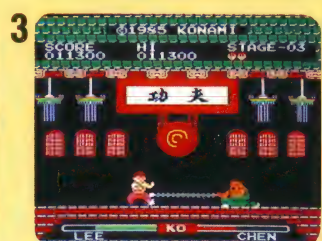
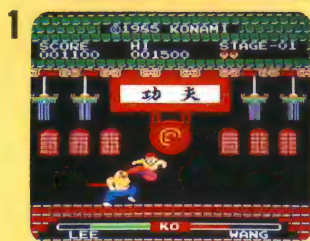
4

①
ほんとびっくり。オリジナルゲームを遊びながらタイプ練習ができてしまいます。
②
ペンギンとカメさんを、コントロールキーで自由自在に動かします。
③
INPUT命令を使った、オリジナルプログラムです。とってもムフフなゲームです。
④
なぜかペンギンのラインダンス。FOR-NEXTを使ったデモプログラムです。

イーアル・カンフー

1

2

3

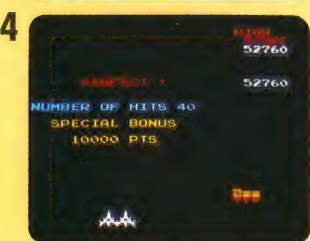
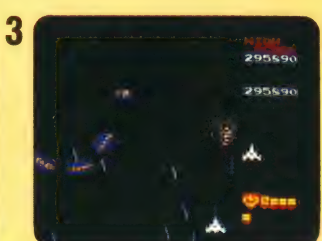
4

①〜④
棒術使い「ワン」、火吹き男「タオ」、くさりガマ「チン」、手裏剣「ラン」、不気味な男「ムー」の５人と闘います。それぞれの必殺ワザを出す前に、ぎりぎりに近ずくと相手はワザを出せなくなります。あとは、飛びげり、足ワザで料理してしまいます。

ギャラガ

1

2

3

4

①
ボスギャラガの出すトラクタービームは、デュアルキャノンファイタへの唯一の道。一面で、デュアルに。

②
ギャラガが編隊を組む前に、３分の２を消すようにします。あとは、ホームであせらず消します。

③
チャレンジ・ステージは、確実にパーフェクトを。パターンを覚えれば、楽々こなせます。

④
20面以後の敵の総攻撃には、シングルファイタで。当然、常にホームにいるようにします。

結極、南極大冒険

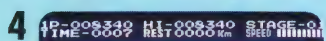
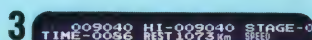
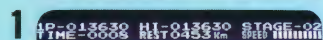
アトランティック、アドベンチャー

1

2

3

4

①

「ペキコプタ」を使って、ひとっ飛び!!いかに上手にペキコプタを使うかが、勝負の分れ目!!

②

南極大陸の地図です。各国の南極基地をまわって、大陸を一周します。

③

クレバスに落っこちゃった。ワァー、出られないよお!!

④

無事、オーストラリア基地へ到着しました。ハタがあがって、大喜びです。

ゼビウス

1

2

3

4

①

ゼビウス空中要塞アンドアジェネシス現れる。画面上方で、8の字を描くようにして避けます。

②

スペシャルフラグは、確実に取りましょう。画面左から9発順番に打っていきます。

③

恐怖のテラジ出現!!とにかくテラジとたて軸を合わせるように。逃げると、必ずやられます。

一生懸命(いっしょうけんめ) ［詳細・明解 パソコンシリーズ］

ファミリーコンピュータ™

ゲーム＋(プラス)ベーシック入門

木村香奈枝・斉藤 千秋 著

●やさしくて,詳しい一生懸命の入門書。

この本は，次のような人に読んでもらえたら…，と思っています。

①ファミリーベーシックを，購入した人に。

②BASIC を，やさしく入門したい人に。

③BASIC を，詳しく入門したい人に。

④ゲームを，よく遊んでいる人に。

この本を書く時，**やさしく,詳しく**書くように心掛けたつもり，です。

「詳しい入門書は良いけど,わかりにくい。」「やさしい入門書は良いけど，物足りない…。」そんな人に，たぶん,本当に向いている本。

やさしく読みたい人は——。

本文だけ，読んでみてください。

本文は,図をたくさん使って,読みやすく,わかりやすくしました。

詳しく読みたい人は——。

本文＋注釈文あわせて，読んでみてください。

やさしい本文で物足りない分，注釈文で満足，できます。注釈文は,スタッフの方が用意してくださったので,詳しく専門的に**大学生以上のレベル**に，合わせてあります。

そして，斉藤千秋と木村香奈枝，の２人で書いた本，です。

一生懸命，ちーちゃんは，ゲームを試しました。

一生懸命，かなえちゃんは，文章を書きました。

文章の書き方も，説明の仕方も，どれもこれも，あまり上手にはできませんでした。**一生懸命**がとりえの本，です。

フォーサイト企画部
斉藤　千秋
木村　香奈枝

●本書の構成。

●1章

この章では，ゲームの攻略法を，説明します。

『ファミリーコンピュータ』用のカートリッジ・ゲームの中から，特に，難しそうな，おもしろそうなゲームを選んでみました。

それぞれのカートリッジ・ゲームの，マップ，登場キャラクタ，テクニックなどを，説明しました。

攻略法を紹介したゲームは，『ギャラガ』，『ゼビウス』『F1·レース』，『バンゲリング・ベイ』，『イー・アル・カンフー』，『けっきょく南極大冒険』の6つです。

●コーヒーブレイク

●2章

この章では，ファミコン，マイコンの基礎知識について，説明します。

『ファミリーコンピュータ』のこと，マイコンのこと，コンピュータのこと，知っておかなくちゃいけない知識，知っていて便利な知識を，やさしく，くわしく説明します。

はじめてマイコンに触れる人のための章，です。

●3章

この章では，ファミコン用BASIC言語『ファミリーベーシック』の使い方を，説明します。

ファミリーベーシックを使って，正統的なBASICプログラミングを，理解できます。キーボードの正しいタイプ方法から，プログラムの構造，BASICの命令を中心に，説明してあります。

●4章

この章では，『ファミリーベーシック』で作った，ゲーム·プログラムの作り方を，説明します。

もくじ

第1章

『カートリッジ・ゲーム』ぜったい攻略法

この章では、『カートリッジ・ゲーム』1億点攻略法を説明します。選んだゲームは、『ゼビウス』、『ギャラガ』、『バンゲリング・ベイ』、『F1・レース』、『イー・アル・カンフー』です。えーと、中でも「ゼビウス」には、とても力を入れて説明しました。なんと、飛行コースまで説明しちゃいましたの。これ以上の攻略法は、ないと思います。ぜんぜん上手になれない人、ぜひ読んでみて下さい。なお、この章は、斉藤千秋が担当しました。

ゼビウス

『ゼビウス』はマップを見ないで遊ぶと，半年。マップを見ると，１か月くらいは十分に遊べる，とっても複雑なゲーム，です。

たぶん，男の子なら，『ファミリーコンピュータ』のゲームの中で，一番おもしろい！て思うはずです。

遊んでみた感想は“むつかしいよぉ”です。

本当に難しくって，すぐ疲れちゃいます。なので，ポーズボタンをちょくちょく押して，休みながら遊ぶのをおすすめします。

わたしは，３週間くらい遊んで，10面ちょっとでした…。

●ゲームの内容

まず，ゲーム内容を説明します。

『ゼビウス』は，16エリアに分かれたゼビウス軍の上空を飛んで，攻撃してくる空中物，地上物を破壊するゲーム，です。空中物には，「ザッパー」で。地上物には「ブラスター」で攻撃します。

『ゼビウス』で，高得点を取るには――

①マップを覚えます。

②ポーズボタンを利用します。

③コントローラを使いやすくします。

④隠れキャラクタを多く発見します。

というようにするとよいみたい，です。

特に，①「マップを覚えます」は，とっても重要です。マップを知らないと，短期間で高得点は取れないと思います。けど，マップを見ちゃうと，『ゼビウス』のおもしろさが半減しちゃうので，みなさんの好きな方を選んでください。『ゼビウス』は，次にどんな攻撃が来るかわからないのが，おもしろいんです。

②「ポーズボタンを利用します」は，手が疲れちゃったり，マップを忘れちゃった時，使うと便利。でも，これも①と同じで，使うとつまんなくなります…。

③「コントローラを使いやすくします」は，コントローラに500円玉とか10円玉を両面テープで付けて，ソルバルウを操作しやすくします。

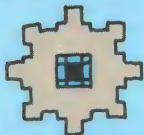
ほんとに短期間で高得点を取りたい人は，マップを見ながら，ポーズボタンをちょくちょく利用しましょう。こうすれば，10面は，すぐに越せる，と思います。

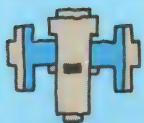
空中キャラクタ

◉トーロイド(30ポイント)

ゼビウス軍の無人機。一度に3～6機現れます。簡単に倒すことができますが，全エリアによく出てきます。ソルバルウとタテ軸が一致すると旋回します。この時に，弾を出すタイプと，出さないタイプがあります。

◉タルケン(50ポイント)

有人戦闘機。2機～5機までが集まって攻撃してきます。ソルバルウが右側にいれば左側から，左側にいれば右側から現れます(左右対称方向)。この時に，弾を発射します。そしてコックピットを反転させて，戻ってゆきます。

◉ゾシー(70、100ポイント)

ゼビウス軍の無慣性飛行物体。ランダムに方向転換する70ポイントタイプと，ソルバルウに向けて方向転換する100ポイントタイプ，バックアタック攻撃をする100ポイントタイプがあります。攻撃はしつこくて，100ポイントタイプは破壊されるまで続く。

◉ジアラ(150ポイント)

トーロイドとだいたい同じ動き，します，でも，かなりのスピードで飛んでくるので，なかなか手ごわいです。3～6機の編隊で攻撃してきます。ソルバルウに向かって弾を発射するタイプと，しないタイプがあります。

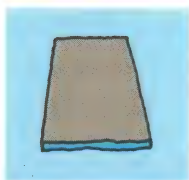
◉カピ(300ポイント)

タルケンの改良型有人戦闘機。最高5機まで現れます。ソルバルウに接近した後，機体を反転させて弾を出しながら，カーブを描いて戻ってゆきます。一応，改良型。でも，それほど，強くないんです。

◉テラジ(700ポイント)

カブトガニ型有人戦闘機。最高5機まで，現れます。カピを，さらに改良した，ゼビウス軍最新戦闘機，です。画面の上から現れて，ソルバルウに接近します。その後，機体をカピと同じように反転させながら戻ってゆきます。でも，カピと違ってその間じゅう弾を出します。

◉シオナイト

ゼビウス軍の反乱分子。ソルバルウのただ1つの味方です。エリア9とエリア14で，登場します。画面左右上方から飛んできて，ソルバルウの前で交差して，その2つがくっついて，戻ってゆきます。

◉バキュラ

エネルギーの集中度を高めた新建材。1画面中に，5枚以上は現れません。このバキュラは，ザッパーで破壊できません。

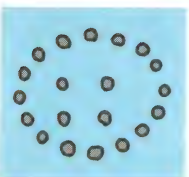
◉ザカート(100～300ポイント)

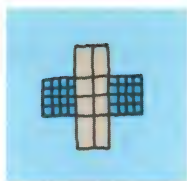
画面上にランダムに現れて，真下に落ちるのと，ソルバルウに向かってくるのがあります。どちらのタイプも，少しすると爆発して弾を発射します。

◉ブラグザカート(600、1500ポイント)

画面上にランダムに現れて，ソルバルウに接近してから爆発します。5発の弾を扇状にばらまきます。ドイントは，600，1500の2タイプあり，それぞれ爆発の条件が違ってます。ザカートの改良型。

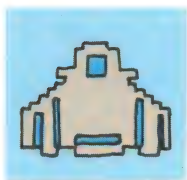
◉ガルザカート(1000ポイント)

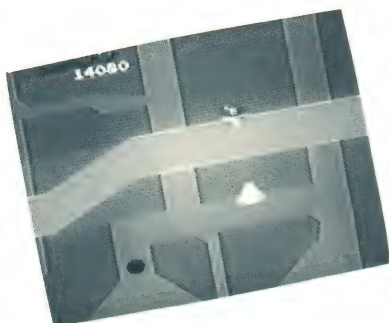
画面上方のランダムな場所から，真下に落ちてきます。画面中央まで落ちると爆発して，16個の弾と4個のブラグスパリオ（誘導弾）を円状に飛ばします。ガルザカートは，バキュラと同じで，現れる場所が決まってます。ブラグガカートの改良型。

◉ブラグスパリオ(500ポイント)

ガルザカートが爆発した時に発射される20個の弾のうちの4つです。これらの弾は施回しながらソルバルウを襲ってきます。ザッパーを当てるとポイントになります。でも，破壊できません。

◉キドスパリオ(10ポイント)

タルケンと同じように，ソルバルウの左右対称方向から超スピードで現れて，ソルバルウをねらって攻撃してきます。ちょっと気をつけないと，やられてしまいます。

◉ブラグザ

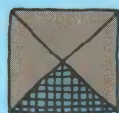
アンドアジェネシスの本体部分。中央のコアが爆発する時，飛び出して，4色に変化しながら画面上方へ消えてゆきます。ザッパーと速度が同じなので，破壊できません。

◉ソルバルウ

プレイヤーが操作するスペースシップです。空中物攻撃用のザッパーと，地上物攻撃用のブラスターが使えます。

地上キャラクタ

◉ゾルバク(200ポイント)

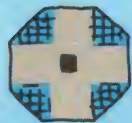
ゼビウス軍の情報収集地上要塞。ソルバルウへは，攻撃してきません。けれども，これを破壊すると，空中物の難易ランクが下がります。

◉グロブター(200～1000ポイント)

有人水陸両用ですから，平地や海の上を自由に動きまわります。

◉バーラ(100ポイント)

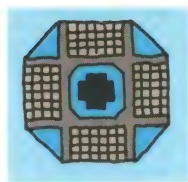
ゼビウス軍のピラミッド型建造物。ソルバルウへの攻撃はしません。ゲームの進行にも，ぜんぜん影響しません。

◉ログラム(300ポイント)

球状のエネルギー弾発射装置。ソルバルウに向けて，同じ周期で，弾を発射してきます。ドモグラムやボザログラムの原形となってます。

◉ドモグラム(800ポイント)

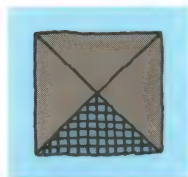
ログラムにホバーリング機能を付けて、どこでも自由に動けるようにしたものです。

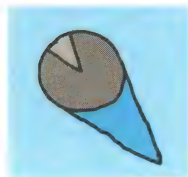
◉ボザログラム(300～2600ポイント)

4個のログラムが連結したものです。中心を一発で倒すと2000点です。周りのログラムを壊すと、600点になってしまいます。

◉デローター(1000ポイント)

画面に現れると、ソルバルウに向けて次々と弾を発射してきます。ログラムに，スパリオの連射機能を付け加えた砲台です。

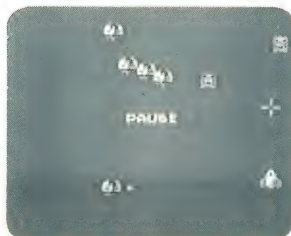
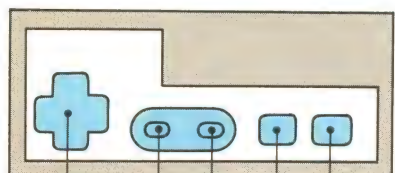
◉ガルデローター(2000ポイント)

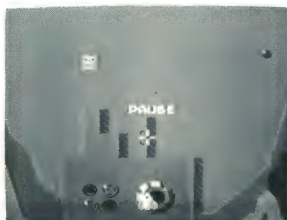
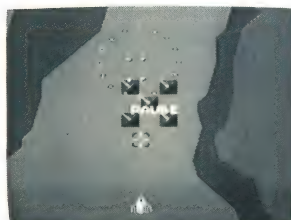
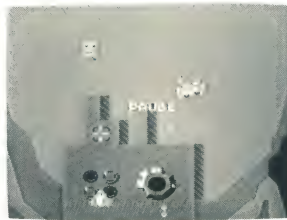
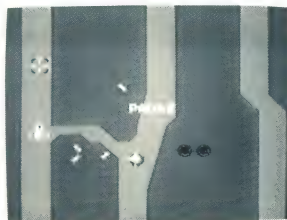
デローターが巨大化したものが，ガルデローターです。下部はバキュラで作られてるので，ブラスターで壊せません。なので，中央部分をねらいます。デローターよりも，数多くの弾を発射します。

◉ガルバーラ(300ポイント)

バーラが巨大化したものが，ガルバーラです。ガルバデローターと同じで，下部はバキュラでできてます。

◉ソル(2000＋2000ポイント)

10エリア中46本が隠されています。普段は，地下に隠されています。ブラスターが撃ち込まれると，現れます。ソルが現れると2000ポイント，それを破壊すると2000ポイント，です。

◉スペシャルフラグ(1000＋1機)

スペシャルフラグゾーンに，横一列ブラスターを撃ち込んでいくと現れる，隠れキャラクタです。現れると1000ポイント増し，通過するとソルバルウが１機増えます。

コントローラの操作方法

このコントローラのままだと，なめらかにソルバルウを動かせないと思います。そういう人は，コントローラの操作ボタンに，500円玉か10円玉を両面テープで張り付けてみましょう。

ブラスター，ザッパーボタンの押し方は，高得点に関係してます。よーく研究して，自分に合った押し方を探しましょう。

エリア1

「エリア1」では，スペシャルフラグと2本のソルを出すようにします。特に，スペシャルフラグは，なるべく出しましょう。

スペシャルフラグを出すには、川の右から9発ブラスターを左端まで順々に打ち込めば大丈夫！

あとは，トーロイドを攻撃しないで，ソルバルウを左右に動かして，トーロイドを逃がしましょう。トーロイドをたくさん攻撃すると，難易度レベルが上がります。

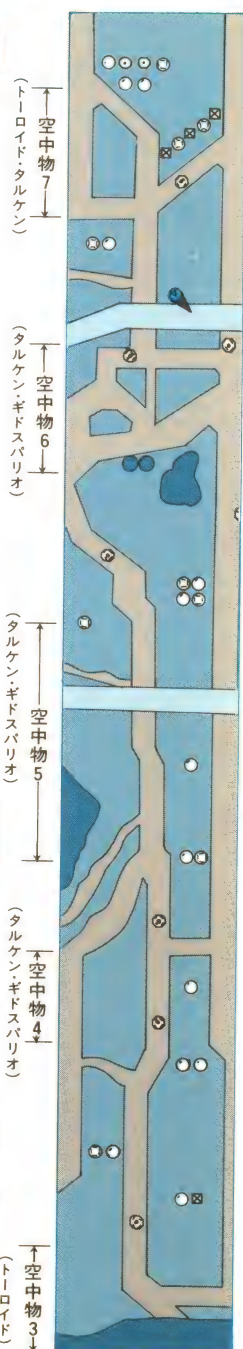
それから，6機あるゾルバクは全部消しておきましょう。

機　　種	得　　点
バーラ	5　(500)
ゾルバク	6　(1200)
ログラム	7　(2100)
グロブターA	3　(600)
ボザログラム	1　(2600)
ソル	2　(8000)
スペシャルフラグ	1　(1000+1機)

エリア2

「エリア2」では，無理をしないでソルを出すことだけ，考えましょう。最後に現れるデロータは，画面に現れたところをブラスターで攻撃します。

10機あるゾルバクは，全部消した方が良いです。

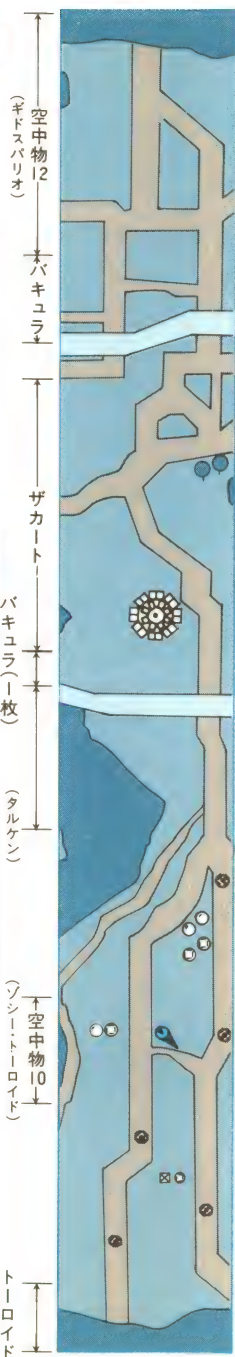
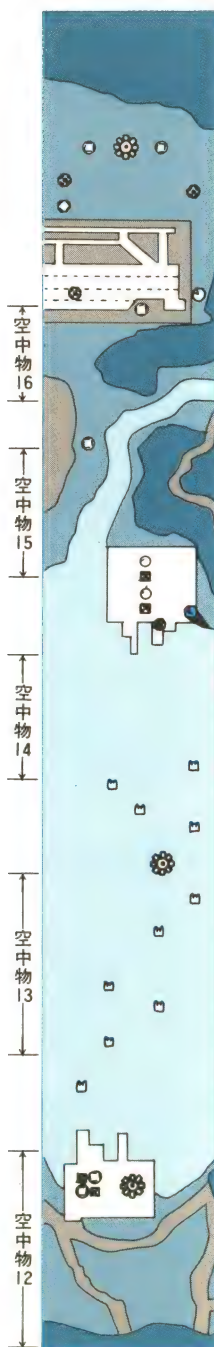
機　　種	得　　点
バーラ	4　(400)
ゾルバク	10　(2000)
ログラム	12　(3600)
ドモグラム	7　(5600)
デロータ	2　(2000)
ソル	2　(8000)

エリア3

「エリア3」は，とっても楽なエリア。
スペシャルフラグのことだけ，考えてくださいネ。最後のガルデロータは，現れたところを攻撃します。横のソルを，忘れないで…。

機　　種	得　　点
バーラ	4 (400)
ゾルバグ	9 (1800)
ログラム	8 (2400)
ガルバーラ	1 (300)
グロブター(C)	1 (600)
ガルデロータ	1 (2000)
ソル	2 (8000)
スペシャルフラグ	1 (1000+1機)

エリア4

はじめてのアンドアジェネシス，です。
「エリア4」は，このアンドアジェネシスが大問題。アンドアジェネシスからの攻撃は，横の移動を中心に8の字を描くようにして，逃げます。この時，ザッパーを連射します。

機　　種	得　　点
バーラ	1 (100)
ゾルバク	4 (800)
ログラム	3 (900)
ドモグラム	5 (4000)
ソル	1 (4000)
アンドアジェネシス	1 (8000)

エリア5

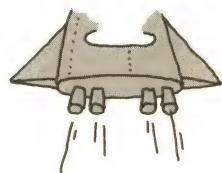
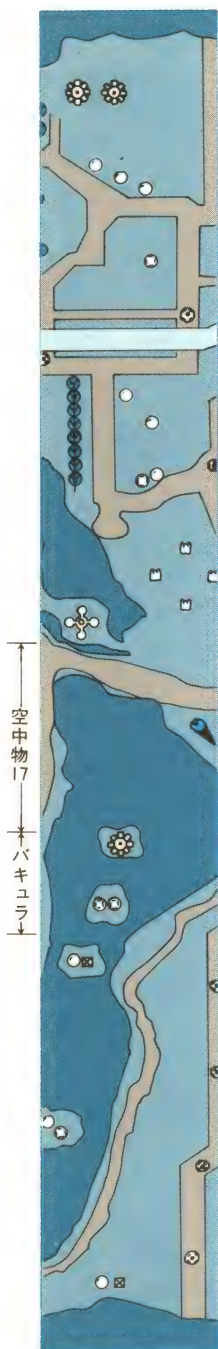
「エリア5」は，これまでのエリアよりも難易度レベルが，かなり上がってます。

最初のガルデローダ，海上のガルデローダは，現れたところを攻撃します。向こう岸の手前にあるスペシャルフラグは，なるべく出しましょう。左から右へ9発ブラスターを順々に撃ち込みます。

港のドモグラムは，早めに倒してくださいネ。空中物の攻撃が始まるから，待ってるとタイヘンです。ついでに，横のソルも出してしまいましょう。

最後のガルデローダも，やっぱり現れたところにブラスターを。

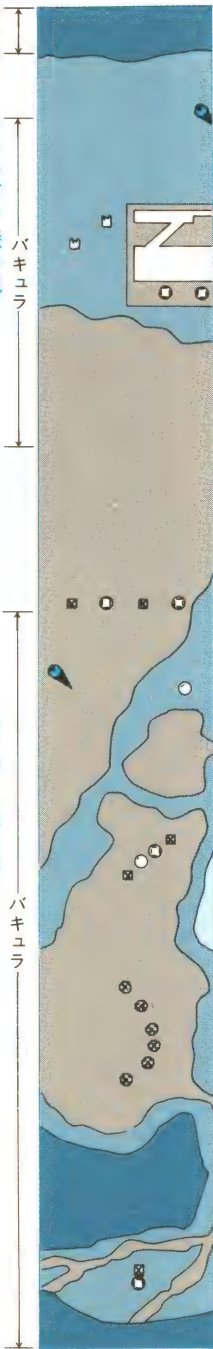
それから，ゾルバクを倒しておくと，後々ラクですね。

機　　種	得　　点
バーラ	4 (　400)
ゾルバク	6 (1200)
ログラム	2 (　600)
グロブダーA	1 (　200)
グロブダーB	7 (2800)
グロブダーC	2 (1200)
ドモグラム	5 (4000)
ガルデローダ	3 (6000)
ソル	2 (8000)
スペシャルフラグ	1 (1000＋1機)

エリア6

「エリア6」は，エリア5よりも簡単，です。

最初のガルデロータをブラスターで倒したら，すぐに下にさがって空中物の攻撃にそなえます。これを注意すれば，あとは寝ててもできます（ほんとはできないですヨ）。

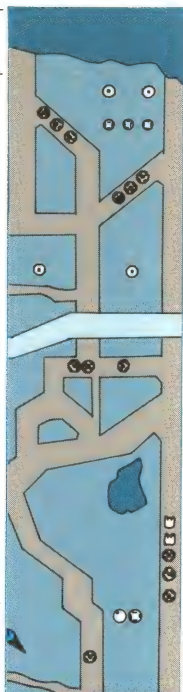
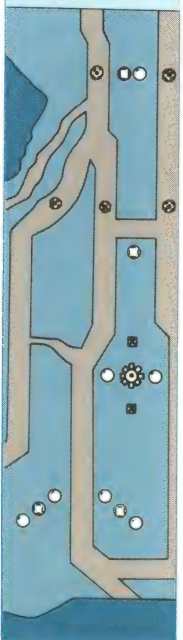
最後に2つ並んだガルデロータは，とても難しそうに見えます。でも，これも，画面に現れたところをねらえば，よいです。

機　　種	得　　点
バーラ	2 (200)
ゾルバグ	5 (1000)
ログラム	9 (2700)
グロブターE	3 (3000)
グロブターG	1 (2000)
ドモグラム	7 (5600)
ガルデロータ	3 (6000)
ボザログラム	1 (2600)
ソル	2 (8000)

エリア7

「エリア7」は，回転ドモグラムへの攻撃とスペシャルフラグが，大切です。特に，スペシャルフラグは最後なので，左から右へ確実に9発打ち込みましょうネ。

回転ドモグラムは，同一円周上をグルグル回ってるから，1か所に照準を固定して倒します。照準は，最下端よりちょっと右上に。

あとは，ソルを2つ出すのと，1万点のスペシャルグロブターにチャレンジ。

機　　種	得　　点
バーラ	5 (500)
ゾルバク	6 (1200)
ログラム	2 (600)
グロブターH	1 (2500)
グロブターI	1 (10000)
ドモグラム	6 (4800)
ソル	2 (8000)
スペシャルフラグ	1 (1000＋1機)

エリア8

「エリア8」は，ドモグラムのエリアです。

ガルデロータを消したあとの空中物の攻撃(こうげき)，ちょっと危険(きけん)です。下がって，攻撃をかわしましょう。

ガルデロータ，デロータは，現(あらわ)れたところを攻撃するのが，よいと思います。

機　　種	得　　点
バーラ	2　(　200)
ゾルバク	8　(1600)
ログラム	8　(2400)
グロブターB	1　(　400)
グロブターH	1　(2500)
ドモグラム	19　(15200)
デロータ	4　(4000)
ガルデロータ	1　(2000)
ソル	1　(4000)

エリア9

「エリア9」は，『ゼビウス』の難問(なんもん)のひとつです。

まず，ソルを4機出します。次に，ガルデロータが現れるので，ソルバルウを上にあげます。ガルデロータが現れたら，すぐにブラスターを撃(う)ち込んで下にさがります。

このガルデロータが，このエリアの難問なんです。

それから，ガルザカートが現れます。ガルザカートは，現れたらスグにザッパーで攻撃するのがよい，と思います。

機　　種	得　　点
ガルデロータ	1　(2000)
ソル	4　(16000)
アンドアジェネシス	1　(8000)

エリア10

「エリア10」は，ドグラム8機のダンス，ガルザカートの3連発，4連デロータという難問が，待ってます。

ドグラムのダンスは，動きをよく見て着実に攻撃するのが，一番です。

5機のバーラが現れる頃に，3連続ガルザカートが出てきます。とにかく，すぐに攻撃してしまいましょう。

8本ソルの所は，照準が〈UP〉に重なるようにして待って，赤く光ったら，ブラスターを次々に撃ち込みます。4本まで，倒せます。

機　　種	得　　点
バーラ	10 (1000)
ゾルバク	5 (1000)
ログラム	1 (300)
ガルバーラ	2 (600)
ドモグラム	9 (7200)
デロータ	4 (4000)
ソル	8 (32000)

エリア11

「エリア11」は，ホッとひと息つけるエリアです。

初めの頃に出てくるグロブダーは，なるべく全部消しましょう。1機で2500点のグロブダーHもあります。

えーと。あとは，ソル6機を出しましょう。

機　　種	得　　点
バーラ	6 (600)
ゾルバク	7 (1400)
ログラム	1 (300)
グロブダーB	3 (1200)
グロブダーF	1 (1500)
グロブダーH	2 (5000)
ドモグラム	5 (4000)
ガルデロータ	1 (2000)
ボザログラム	1 (2600)
ソル	6 (24000)

エリア12

「エリア12」は，慣れればカンタン，慣れないと困難なエリアです。

4本ソルを出すには，照準を〈1 up〉のあたりに移動させて待つのがよいみたい…。

慣れない頃は，ポーズして一時停止してマップを見ながら，チャレンジ。

機　　種	得　　点
バーラ	6　(　600)
ゾルバク	8　(1600)
グロブダーB	4　(1600)
グロブダーC	3　(1800)
グロブダーF	1　(1500)
ソル	4　(16000)
ドモグラム	13　(10400)
デロータ	9　(9000)
ガルデロータ	1　(2000)

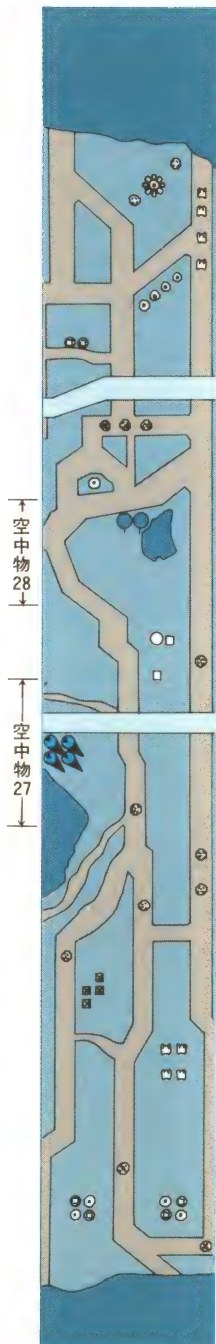
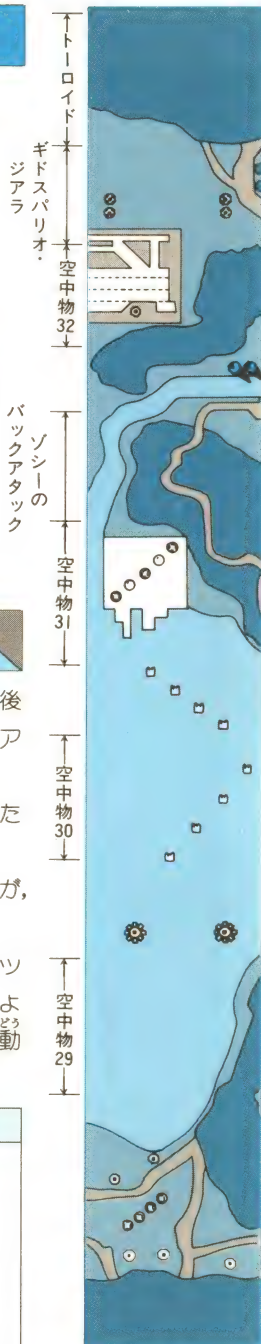
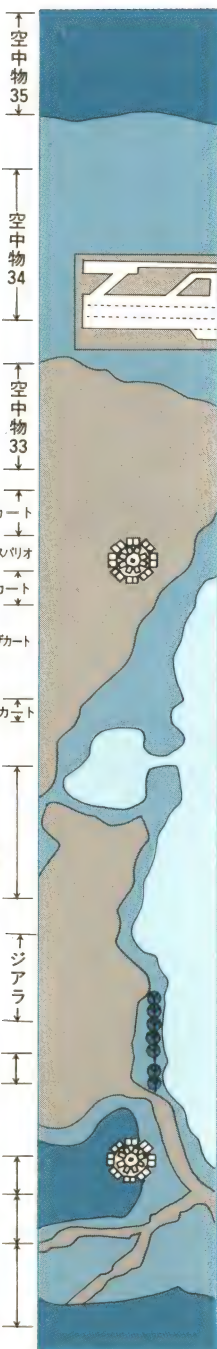
エリア13

「エリア13」は，初めの5機のデロータ，後半のゾシーのバックアタックが危険なエリアです。

5機のデロータは，画面上で待って現れたところを次々に攻撃します。

港のログラムは，早めに倒しておいた方が，よいみたいです。

ゾシーのバックアタックには，バックアタックをされる前に，その後方にまわり込みましょう。この時は，なるべく円を描くように移動します。

機　　種	得　　点
ゾルバク	7　(1400)
ログラム	2　(600)
グロブダーB	6　(2400)
グロブダーH	2　(5000)
ドモグラム	4　(3200)
デロータ	6　(6000)
ガルデロータ	2　(4000)
ソル	2　(8000)

エリア14

「エリア14」は，アンドアジェネシスが２回も現れるエリアです。

ブラグザカート連発と，２機目のアンドアジェネシスが，難関だそうです。ブラグザカートは，出てきたら攻撃しましょう。間に合わなかったら，画面の端に移動して，スパリオをよけましょう！

機　　種	得　　点
アンドアジェネシス	2　(14000)

エリア15

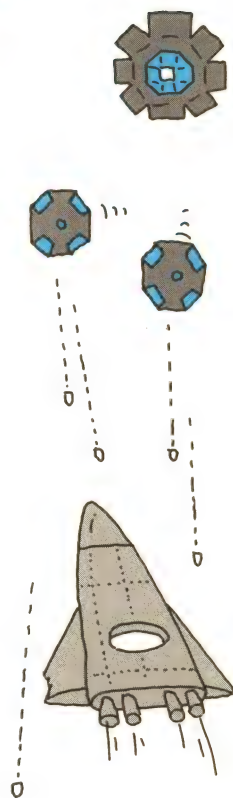
「エリア15」は，４機のドモグラムのダンスと最後のデロータ＋ガルデロータの所が，難しいそうです。

機　　種	得　　点
ゾルバク	9　(1800)
ログラム	6　(1800)
グロブターB	2　(　800)
グロブターH	3　(7500)
ドモグラム	14　(11200)
デロータ	4　(4000)
ガルデロータ	2　(4000)
ソル	4　(16000)

エリア16

「エリア16」は，『ゼビウス』最後のエリアです。

ドモグラム＋デロータのある前半，6機のログラムとガルデロータの中半，4機のデロータのある後半が，クライマックスなんだそうです（わたし，ここまできたことないので，スタッフの方に聞きました）。

機　種	得　点
ゾルバク	5　(1000)
ログラム	10　(3000)
グロブターH	5　(12500)
グロブターI	1　(10000)
ドモグラム	13　(10400)
デロータ	8　(8000)
ガルデロータ	3　(6000)
ソル	4　(16000)

ギャラガ

『ギャラガ』が，今回遊んだゲームの中では，**一番好き**！になりました。

2番目は『F1・レース』。3番目は『ゼビウス』…。

こんなふうに並べると，わたしは，あんまり一生懸命じゃないのが，好きみたいです。『ゼビウス』みたいに，すごく複雑なゲームだと，遊んでるのか，勉強してるのか，わからなくなっちゃうもん。

その点，『ギャラガ』は，一生懸命にやらずに楽しめるゲーム，だと思います。

●ゲームの内容

まず，ゲーム内容を簡単に説明します。

『ギャラガ』は，ファイターを左右に操作して，凶暴なエイリアン『ギャラガ』を倒すゲーム，です。『ギャラガ』は，**ミサイル、体当たり、編隊飛行**などの攻撃をしてきます。ボスギャラガは，トラクタービームを発射します。このビームをファイターが受けると，捕虜になってしまいます。捕虜になったファイターを取り戻すと，デュアルファイターになり，攻撃力が倍になります…。

『ギャラガ』で，高得点を取るには――

①ファイターのミサイルを多く撃ちます。
②デュアルの時でも，ミサイルを避けられるようにします。
③攻撃パターンを覚えます。
④デュアルに，すぐになります。

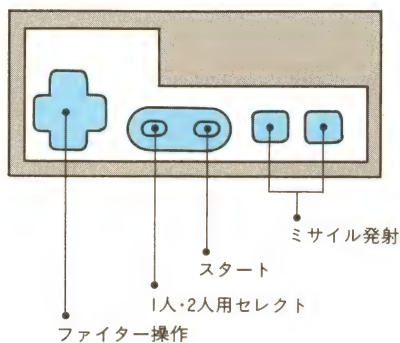
という4つのことを，なるべく守るようにすると，よいみたいです。

それから，特殊ギャラガは，深追いしない方がよい，と思います。特殊ギャラガに気を取られているうちに，ブルーギャラガの**バックアタック**でやられること，よくあるんです。気をつけて，くださいネ。

あとは，残りのギャラガが3匹，なんて時にも，気をつけます。この時も，とってもやられやすいです。なんとなく，クリアしたつもりになっちゃうし，それに，スグ消えそうって深追いしやすいし…。なので，こういう時は，ホームポジションにいて，ギャラガが近くを通るのを待つ，のが良いと思います。

わたしは，だいたい**10日くらい**で，32面をクリアできました。

あんまり一生懸命に遊ばないで，楽しく遊びましょうネ。たかがゲーム，だもん。

コントローラの操作

・SELECT ボタンか START ボタンを押すとセレクト画面に移ります。SELECT ボタンで，１人・２人用を選んでから，START ボタンを押してください。ゲームが始まります。

・ゲームの途中で START ボタンを押すと実行が一時中断されます（ポーズ機能）。もう一度押すと再開します。

・リセットボタンを押しても，最高得点は消えません。

登場キャラクタ

ファイター

デュアル
ファイター

◉ファイター

「ファイター」は，プレイヤーの操作するスペースシップ（宇宙船）のことです。自在な左右移動が得意で，ミサイルを連射して「ギャラガ」を倒します。

ファイターのミサイルは，一画面に２発しかうてません。

「デュアルファイター」の時は，２発並列にミサイルを発射できます。

ボスギャラガ

◉ギャラガ

正体不明の昆虫型エイリアン。とても，凶暴なのだそうです。編隊飛行をして，次々にカーブを描いて攻撃してきます。

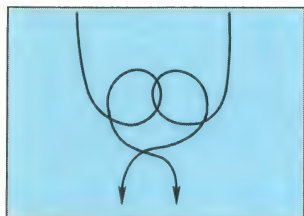
「ボスギャラガ」は，編隊の一番上に４匹並んでいます。ボスギャラガは，２発命中させないと倒せません。

ボスギャラガは「トラクタービーム」を発射します。このトラクタービームをファイターが受けると，捕虜になってしまいます。

攻撃中のボスギャラガを消すと，敵の攻撃は全体的に弱くなってしまいます。

レッドギャラガ

ブルーギャラガ

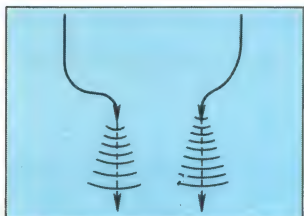
ギャラガの行動パターン

◉ボスギャラガ

（Aパターン）

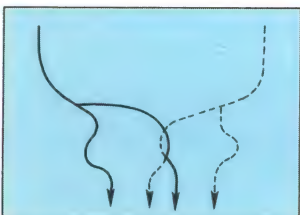
画面中央で一回転してから，攻撃してきます。

回転中か，ファイターに近づいた頃が，撃ちごろ。

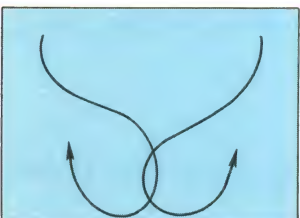
けれども，捕虜がいる場合は，回転中じゃなくて，ファイターに近づいた頃に倒す方がよい，と思います。

（Bパターン）

画面中央でストップして，「トラクタービーム」を発射します。

まちがって「トラクタービーム」受けた時は，ボスギャラガをねらってミサイルを連射しましょう。上手になると，2回に1回くらい助かるようになります。

面数が進んでる時は，クルクル回ってる時でもミサイルを発射して，少しでも倒すようにします。

◉レッドギャラガ

画面の最上部と中央と最下部で，方向を変えて攻撃します。

あまり，近づいて撃たない方がよいんですネ。

◉ブルーギャラガ

画面の最下部で一回転して，攻撃してきます。このブルーギャラガにやられること，多いです。特に，一回転して後ろから。攻撃（バックアタック）に気をつけてください，ね。

ギャラガの得点

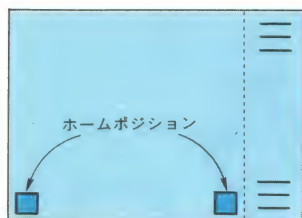
名　前	レッドギャラガ	ブルーギャラガ	ボスギャラガ		
型					
			単独	2機	3機
待機中	50	80			
攻撃中	100	160	400	500	1600

特殊ギャラガの得点

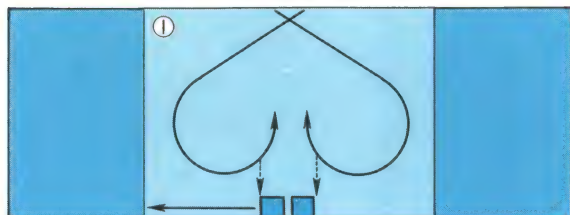
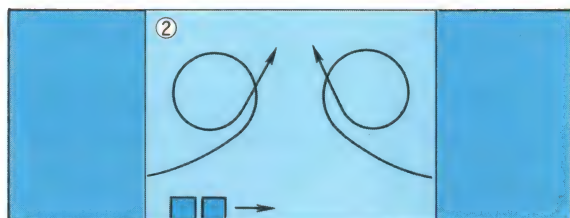
名　前	サソリ	ボスコニアン	ギャラクシアン
型			
得　点	1000	2000	3000

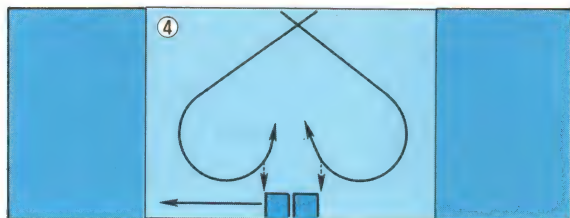
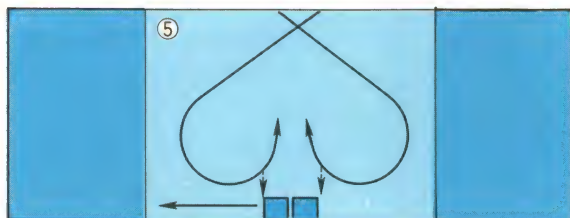
ギャラガの「プロ・テクニック」

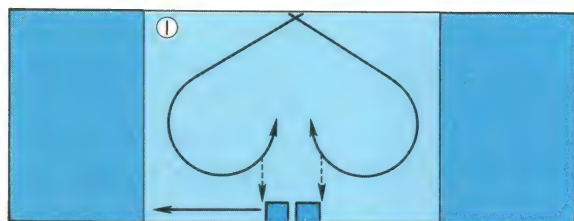
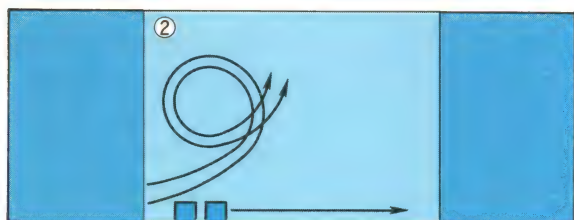
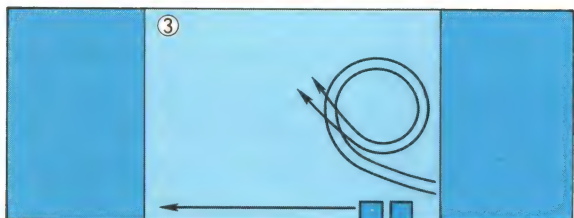
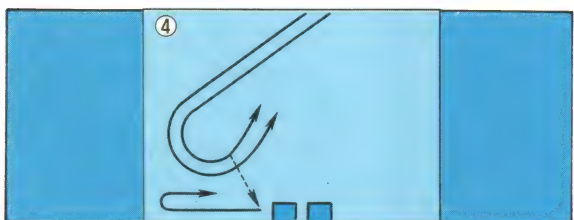
	内　　容
1	「ギャラガ」が編隊を組む前に，消します。
2	なるべく，デュアルファイターで。
3	攻撃パターンを覚えます。
4	攻撃のはげしい時は，画面下半分だけ見ます。
5	ファイターの「ホームポジション」を決めておきます。
6	左右にこまかく動いて，攻撃します。
7	無理をしない。

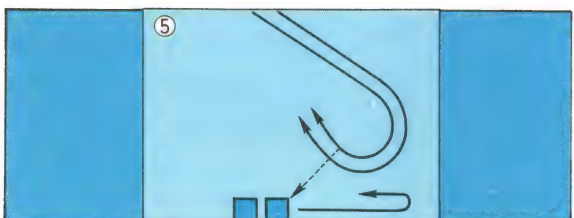
ホームポジション

「ホームポジション」は　攻撃のはげしい時，常にいるようにする場所のこと，です。画面の右端，左端のどっちでも，好きな方でよいです。

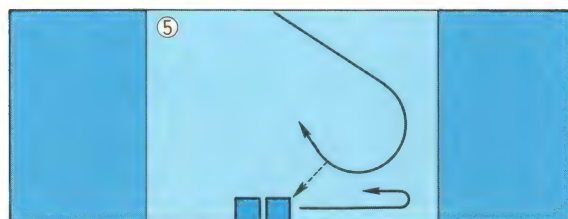
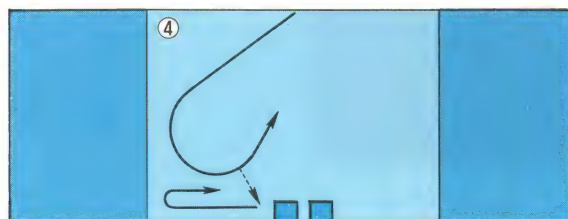
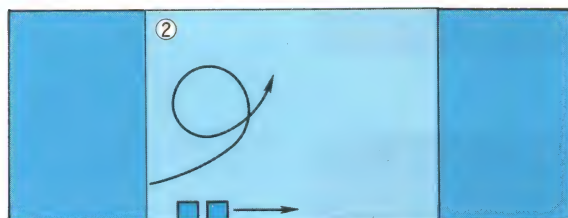
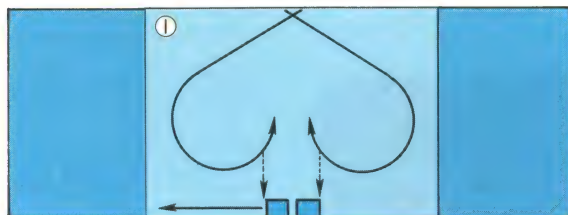
このホームポジションにいると，ギャラガの攻撃は，必ず片方からだけになります。

Aパターン

① 攻撃メンバー→レッド＋ブルー
プロ・テク→中央で待機して，ギャラガが現れたところを「シンクロファイト」で攻撃します。
次に，左側へすばやく移動して，残りのギャラガを消します。

② 攻撃メンバー→ボス＋レッド
プロ・テク→左側から現れる「ボスギャラガ」をなるべく全部消します。けれども，面数が浅い時は，ボスギャラガを消さずに高得点をめざすのも，よいと思います。

③ 攻撃メンバー→レッド
プロ・テク→まず，左側から現れるレッドギャラガを全部消してしまいます。次に右へ移動して右側から現れたのも，全部消します。

④ 攻撃メンバー→ブルー
プロ・テク→①と同じように，まず中央で待機して，ギャラガが現れたところを「シンクロファイト」で攻撃します。そして，左側に移動して残りのブルーギャラガを全部消してしまいます。

⑤ 攻撃メンバー→ブルー
プロ・テク→①，④と同じです。
① ④ ⑤共に，ギャラガが体当たりしてくることがあるので，注意してくださいネ。よく，これでやられちゃいます。

Bパターン

① 攻撃メンバー→レッド＋ブルー
プロ・テク→「Aパターン」の①と同じです。

② 攻撃メンバー→ボス＋レッド
プロ・テク→左側からボスギャラガとレッドギャラガが現(あらわ)れます。なるべく全部消しましょう。左端から画面中央まで，「シンクロファイト」で撃(う)ち続けます。

③ 攻撃メンバー→レッド
プロ・テク→②と同じように攻撃します。

④ 攻撃メンバー→ブルー
プロ・テク→中央で待機して，「シンクロファイト」でミサイルを撃ちます。残りのブルーギャラガは左側へ移動して消します。

けれども20面以降はなるべく深追いは，やめましょう。ギャラガの体当たりもあります。

⑤ 攻撃メンバー→ブルー
プロ・テク→④と同じように攻撃します。

Cパターン

① 攻撃メンバー→レッド＋ブルー

プロ・テク説明→Aパターンの①，Bパターンの①と同じです。ギャラガが現れる前に，ミサイルを「シンクロファイト」で発射して，左側へ移動して，残りのギャラガを消します。

② 攻撃メンバー→ボス＋レッド

プロ・テク説明→左側から現れる，ボスギャラガとレッドギャラガを「シンクロファイト」で，やっつけます。面が進むほど，ボスをここで消した方がよいみたいです。

③ 攻撃メンバー→レッド

プロ・テク説明→右側からレッドギャラガが現れるので，なるべく全部消しましょう。

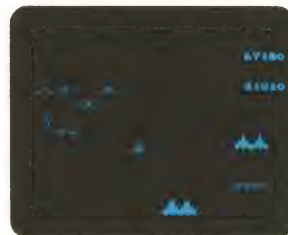
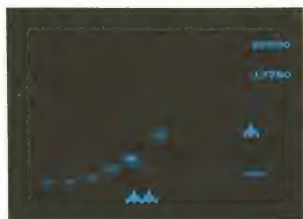
④ 攻撃メンバー→ブルー

プロ・テク説明→中央で待機して，ギャラガが現れる前に「シンクロファイト」でミサイルを発射します。残りのギャラガは左へ移動して消しましょう。

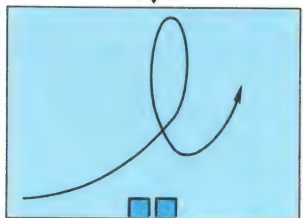
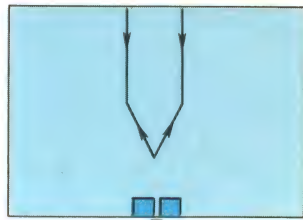
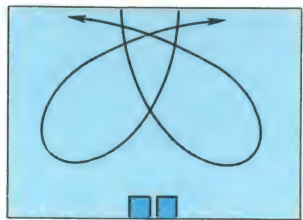
ただし，20面以降は深追いはしない方がよいです。

⑤ 攻撃メンバー→ブルー

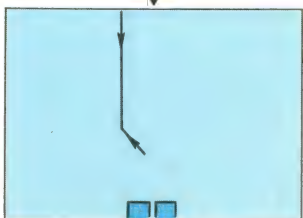
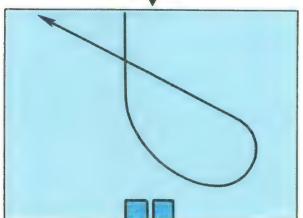
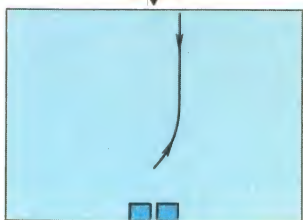
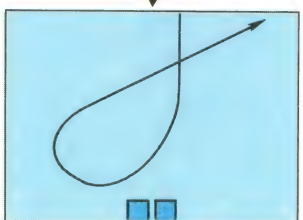
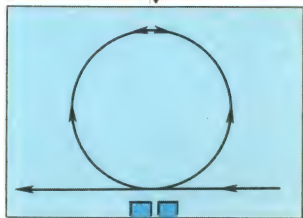
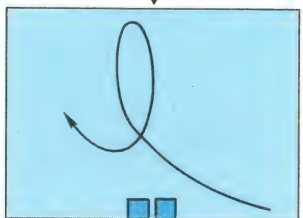
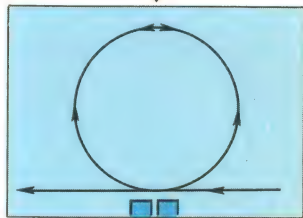
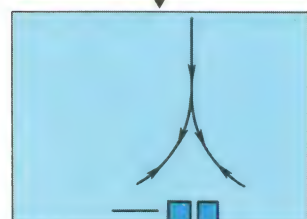
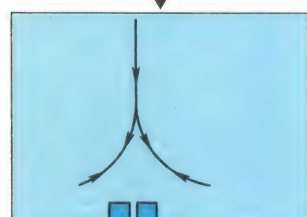
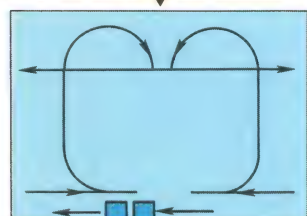
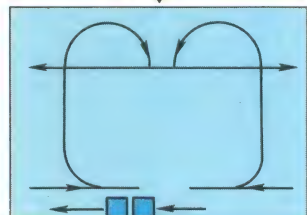
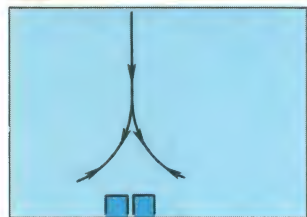
プロ・テク説明→④と同じ，です。

パターンの難易度

Aパターン
>
Bパターン
>
Cパターン
Bパターン
Cパターン
チャレンジング・ステージ
Aパターン
Bパターン
Cパターン
チャレンジング・ステージ

チャレンジング・ステージ3

デュアルファイターを画面中央に，移動(いどう)させます。あとはひたすら，ミサイルを連射(れんしゃ)するだけ，です。パーフェクトは，確実(かくじつ)に取れます。

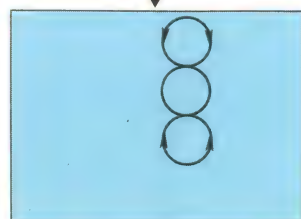
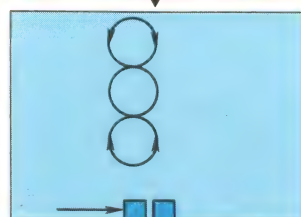
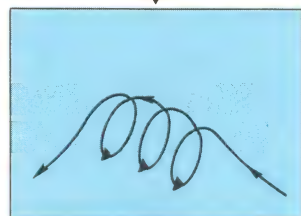
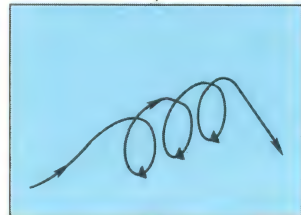
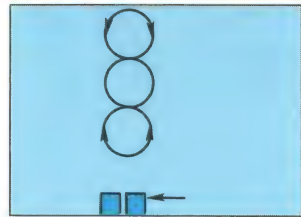
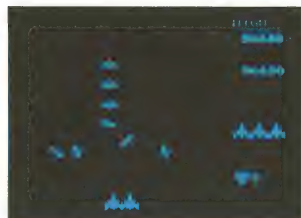
普通(ふつう)のファイターでもラクラク，パーフェクトです。

チャレンジング・ステージ7

ステージ3と同じように，デュアルファイターを画面中央に移動させます。タイミングを計って，ミサイルを発射します。

普通のファイターでは，ちょっと難(むずか)しい，と思います。

チャレンジング・ステージ11

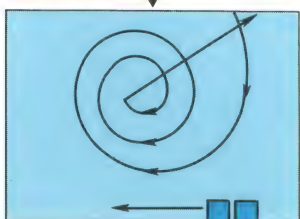
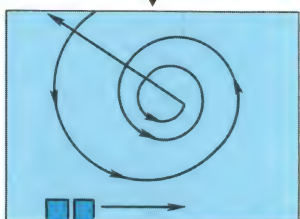
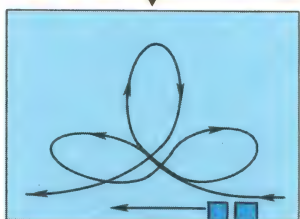
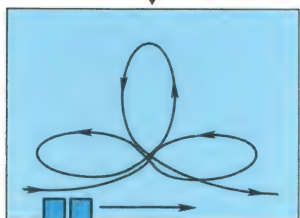
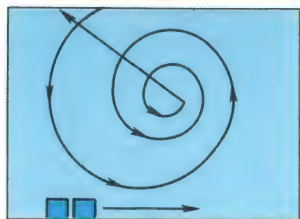
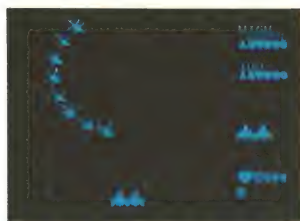
中央よりも少し左へ移動して上から降りてくるトンボを消します。

ボスが，左下から現れますので，まず，ボスから消します。次に，右へ移動して残りのトンボを消します。

トンボの編隊が，左下と右下から現れます。まず，左からのトンボを消して，右からのトンボが上から降りてくるところを消します。

チャレンジング・ステージ15

最初のサソリの編隊は，円の中心にファイターを移動させて，消します。

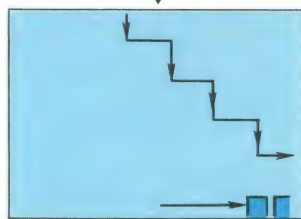
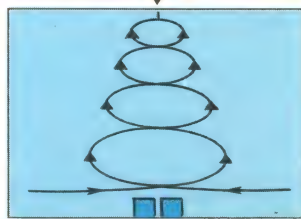
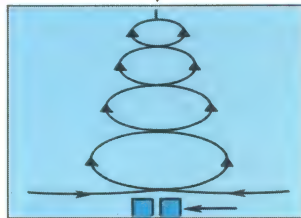
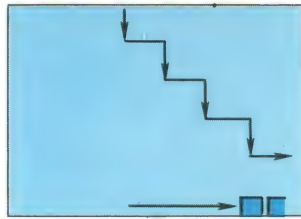
次のボス＋サソリ編隊は，同じ場所で消します。残りは，右へ移動して消してください。

右からのサソリ編隊も同じようにして，消してしまいます。

チャレンジング・ステージ19

フウシャ編隊が，ぐるぐる回りながら現れます。デュアルファイターを左に移動させて，ミサイルを撃ちながら右へ移動します。

ボス＋フウシャ。編隊は，デュアルファイターを左へ移動させて，右へ動しながら撃ちます。

次のフウシャ編隊も，同じように攻撃します。

フウシャの編隊が，2回現れます。パターンは　最初のフウシャ編隊と同じです。

チャレンジング・ステージ23

初めに現れるボスコニアン編隊は，右端で待って撃ちます。

または，中央で待って，右へ移動しながら撃ちます。

ボス＋ボスコニアン編隊は，中央に移動して撃ちます。

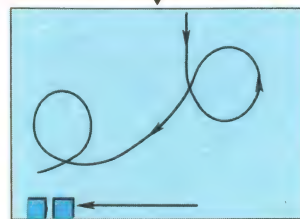
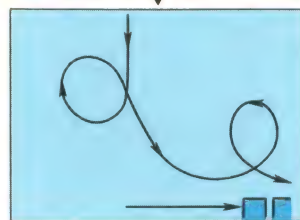
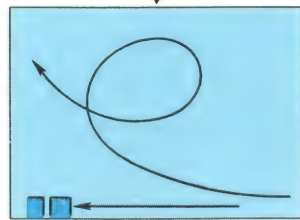
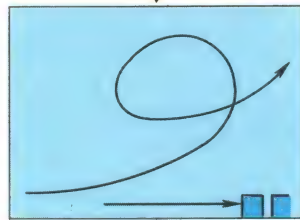
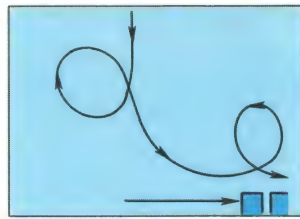
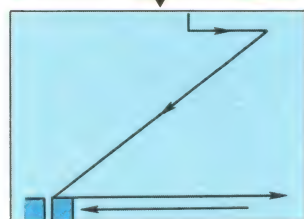
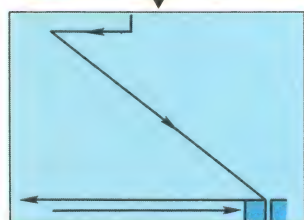
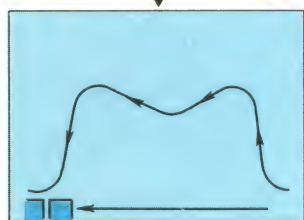
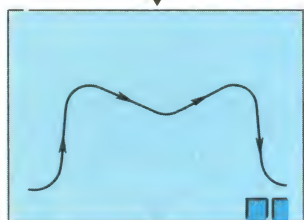
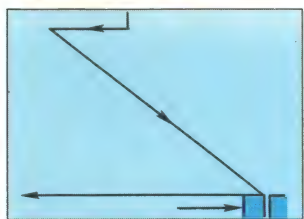
次のボスコニアン編隊も，同じように，倒します。

ボスコニアン編隊が，2度，初めのボスコニアン編隊と同じように現れます。同じようにして，消しましょう。

チャレンジング・ステージ27

最初のギャラクシアン編隊は，右端に移動して，倒しましょう。ボス+ギャラクシアン編隊は，そのままの位置で待ちぶせて，撃ちます。

次のギャラクシアン編隊も，同じようにして，待ちぶせて倒します。

ギャラクシアン編隊が，2度，最初のギャラクシアン編隊のように出現します。画面の右端，左端，それぞれに移動して倒します。

チャレンジング・ステージ31

最初のエンタープライズ編隊は，デュアルファイターを中央より少し左から，右へ移動しながら撃ちます。

ボス+エンタープライズ編隊は，左端から右へ移動させながら撃ちます。

次のエンタープライズ編隊も同じようにして，消します。

エンタープライズ編隊が，2度，最初のエンタープライズのように現れます。中央より少し左から右へ，中央より少し右から左へ，移動して倒します。

F1·レース

『F1・レース』は，おもしろいゲーム，です。
わたし，今回遊んだ中では，『ゼビウス』，『ギャラガ』，そして，この『F1·レース』をおすすめしますっ!
このゲーム，初めの頃は，ぜんぜん上手にできませんでした。うーん、アクセルとブレーキの使い方が，よくわからなかったんです。慣れると，自然と動かせるようになりました。
わたしの場合は，1週間くらい遊んで9コースクリアのペース，です。どうしても，10コースはクリアできませんでした。もう，残念ですぅ。
でも，やっぱり，女の子向きのゲームじゃないですよ。

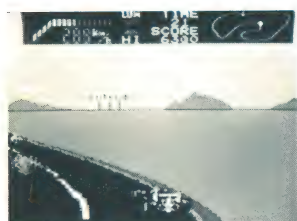
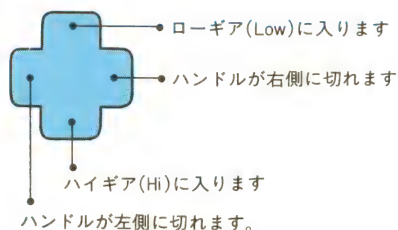
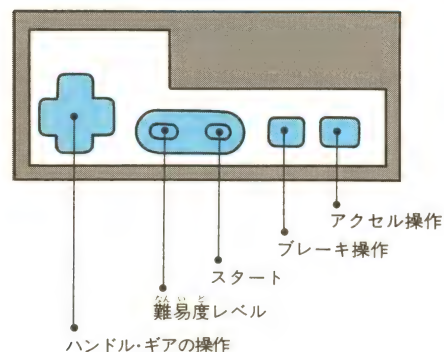
●ゲームの内容

ゲームの内容を，簡単に説明します。
『F1·レース』は，F1·カーを操作して，コースを時間内に2周するゲーム，です。ひとつのコースをクリアすると，次のレベルのコースへ移ります。コースは，全部で10コースあります。
『F1・レース』で，高得点を取るには——
①コースを覚えます。
②なるべく，まっすぐ走るようにします。
③カーブする時，あまり音がしないようにします。
というようにするとよいみたい，です。
本物のレーサーは，コースを歩いて覚えたりするそうです。でも，歩いたりはできないので，よーく覚えるようにしましょうネ。コースを覚えてないと，スピードを出し過ぎたりしちゃいます。
とにかく，他の車にぶつからないようにするのが，基本，です。特に，コーナーでの無理な追い越しは，やめた方がよいと思います。
これだけ，気をつければ8コースまでは，クリアできます。でも，9コース，10コースになると，これだけじゃあ無理。まだまだ，走り方にムダがあります。見つけて直しましょう。ビデオを使うと，よいみたいです。

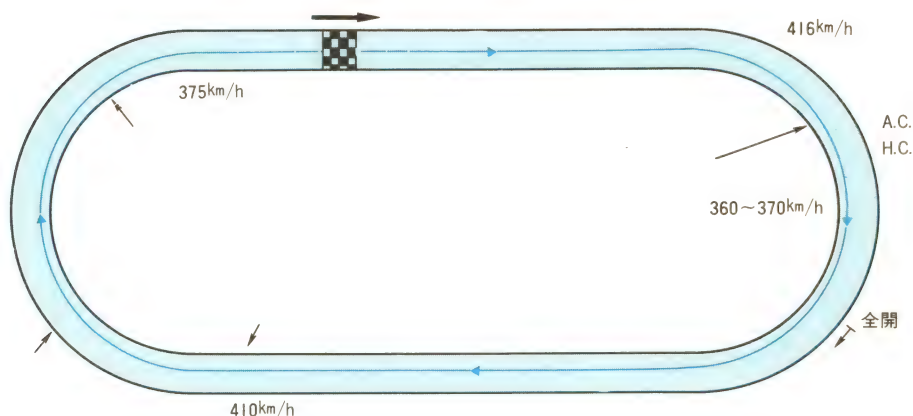
他車と接触しても大丈夫

クルマが，中央線の真上にいる時と，タイヤを半分くらい路肩に乗せてる時は，他車と接触しても，クラッシュしません。

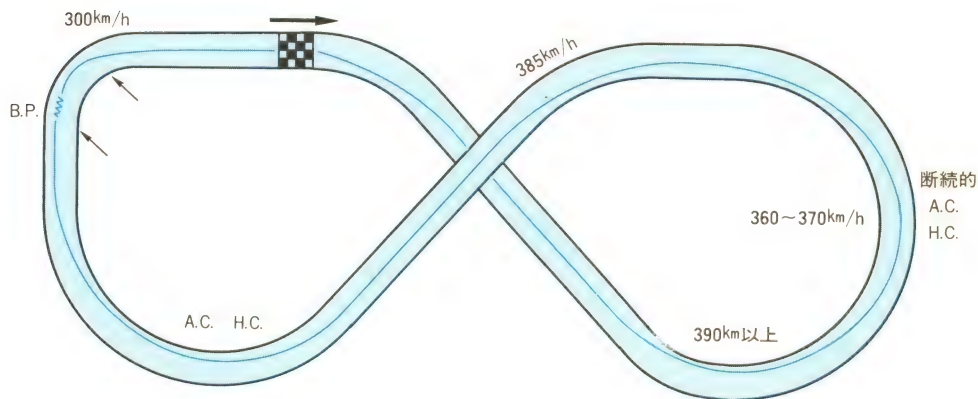
◉中央線の真上の時

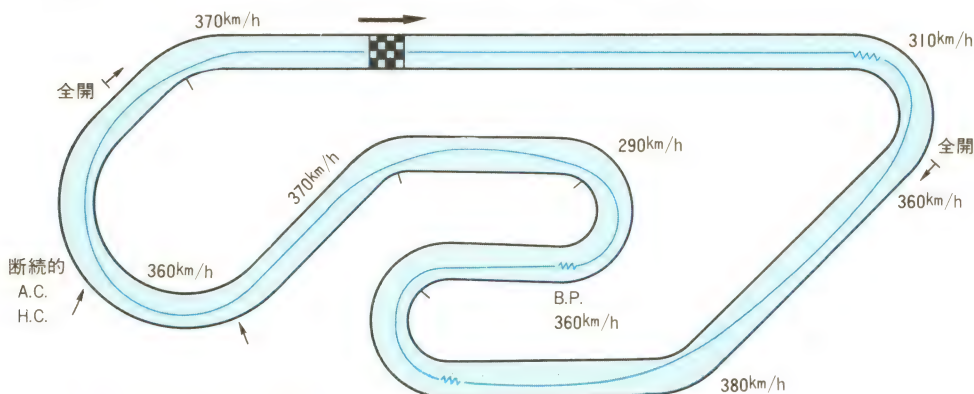
◉路肩にタイヤを半分乗せている時

コントローラの操作方法

1コース

「1コース」は，無理をしないで走れば楽にクリアできます。スピード調整は，ブレーキを使わないで，アクセルを離して行います。

1コースみたいな大きなコーナーを曲る時は，アクセルを一瞬(いっしゅん)離(はな)してハンドル操作(そうさ)をすると，よいみたい。

2コース

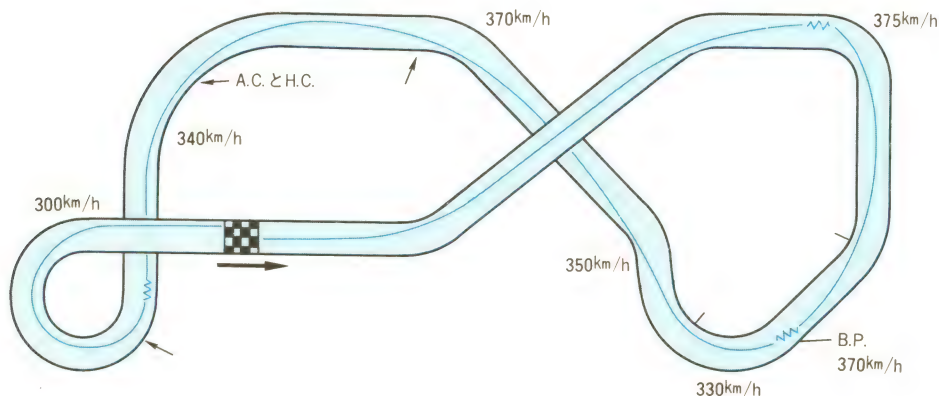
1コースでは，ブレーキを使いませんでした。この「2コース」では，最後の「L字コーナー」で使います。

L字コーナーでは，無理をしないで曲るのもよい，と思います。このコーナーでは，2回くらいならクラッシュしても，平気です。

3コース

「3コース」は，1コースや2コースのようには走れない，と思います。それは，S字コーナーがあるからです。S字コーナーでは，それぞれのコーナーの入口で，ブレーキをかけます。あとは，上手に，アクセルを操作(そうさ)してスピード調整します。コーナーを曲っている時に，キキッ，という音が鳴るか鳴らないか位のスピードが，グッド!!S字コーナーを抜(ぬ)けた後のU字コーナーは，テールが流れないように，スピードコントロールしましょう。

4コース

「4コース」は，後半の270°コーナーが難関(なんかん)です。

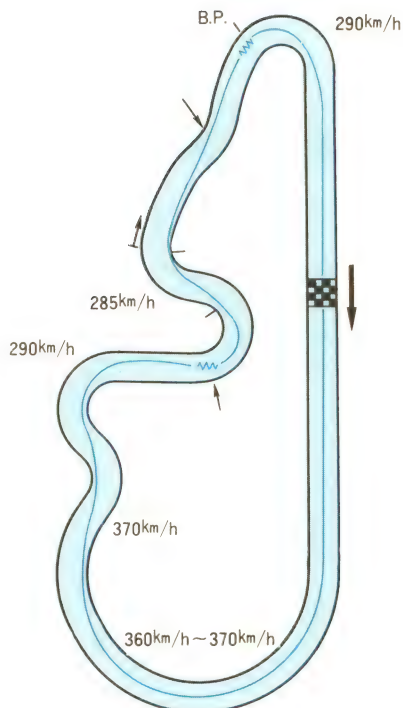
この270°コーナーは，ほんとうは，コーナー入口まで全速力で，急にブレーキングして減速(げんそく)します。でも，初めは上手にできない，と思います。そういう時は，無理をしないで，コーナーの前で減速をしておきましょう。

それぞれのコーナーでは，他の車にぶつからないように注意します。

5コース

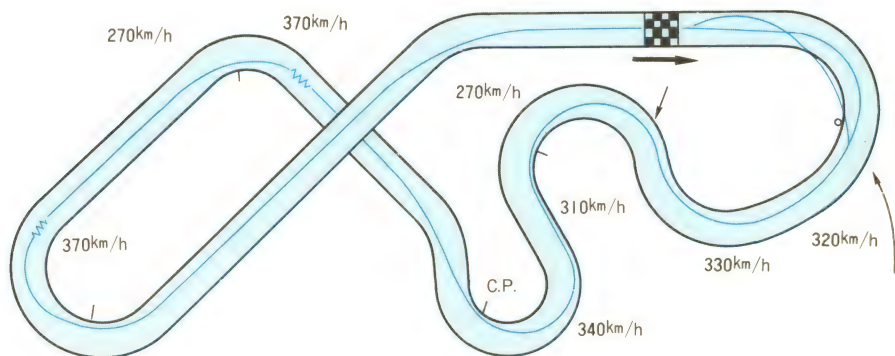
「5コース」は，1〜4コースの難関(なんかん)を一つにまとめたコース，です。

特(とく)に，中盤では連続してコーナーが現(あらわ)れます。ここで，マップをよく覚えていないと，自分がどこを走っているかわからなくなってしまいます。マップをよく覚えてから，このコースは走りましょう。

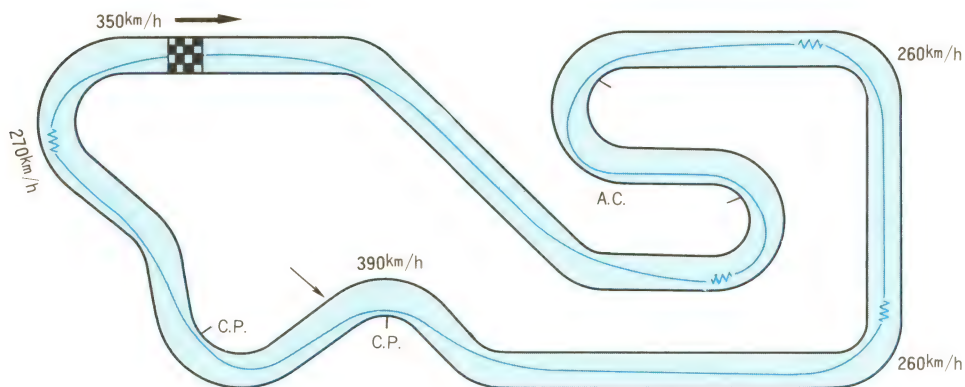
もう1つの難関は，後半のヘアピンカーブ，です。

かなり減速(げんそく)しないと，ポールにぶつかってしまいます。

6コース

「6コース」になると，マップを頭で覚えていてはクリアできません。体で覚えるように，何回もトライしましょう。

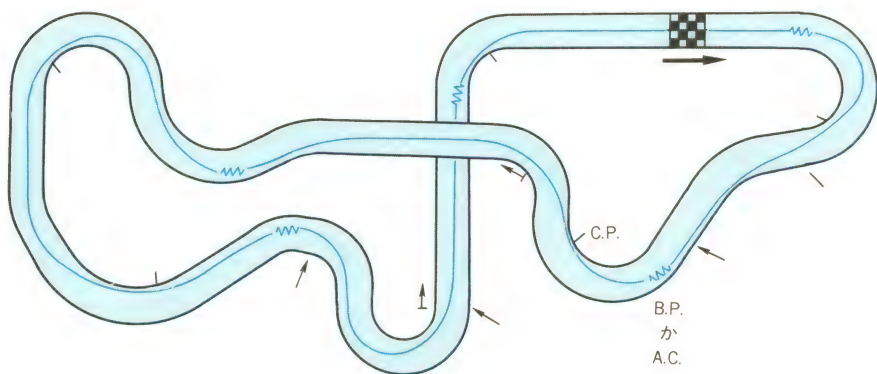
難(むずか)しいのは，序盤から中盤のヘアピン，S字コーナーです。タイヤが鳴かないように，気をつけて走ります。最後のヘアピンは，コースから飛び出さないように，一気に減速。他の車にぶつからないように，注意しましょう。上手になると，一度クラッシュしても2周できます。2周目はスピードが出てますから，最初のヘアピンでは十分に減速しましょう。

7コース

「7コース」は，6コースをちょっと難しくしただけに見えます。でも，走るとわかると思います。6コースに慣れた感覚で走ると，スピードを出し過ぎてクラッシュしてしまいます。

前半のS字コーナー，後半のU字コーナーでは，スピードを出し過ぎないように気をつけましょう。

8コース

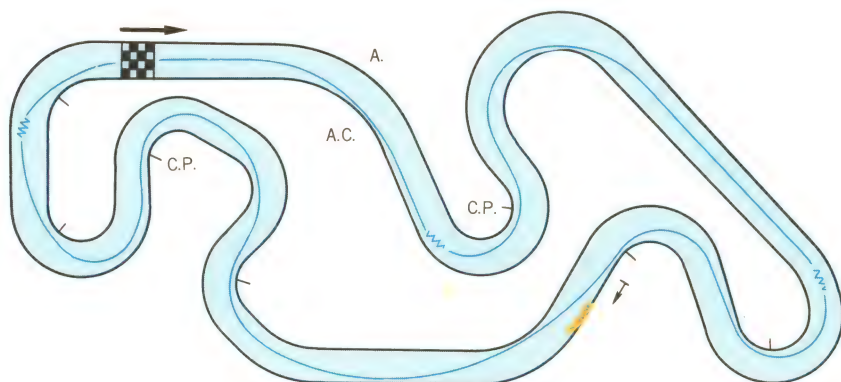
「8コース」には，コレという難関がないです。これまでのコースをクリアできた人には，それぞれのコーナーは無理なく抜けられる，と思います。

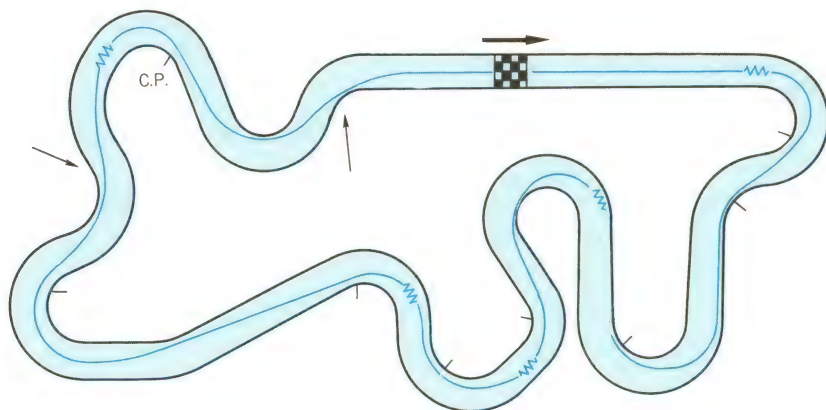
ところが，無理なく抜けたはずでもタイムアウトになってしまいます。

この「8コース」以降は，どうやってタイムを短縮させるかが問題です。

9コース

「9コース」は，とっても難しいコースです。

かなり無理に走ったつもりでも，タイムアウトになってしまいます。それは，コーナーなどでのコースの取り方があまいから，です。どうしても，タイムアウトになっちゃう人は，コースの取り方を，考え直しましょう。

もちろん，クラッシュ＝(イコール) ゲームオーバー，です。

10コース

「10コース」は，わたしもまだクリアしたことないので，みなさんの参考になるようなこと，言えないです。ごめんなさい。アウト・イン・アウト走法を，どれだけ利用するかが，クリアへのカギだと思っています。

バンゲリング ベイ

『バンゲリング ベイ』は，完全に男の子向きのゲーム，です。女の子には，ぜんぜん向いてない，と思いました。

なぜって，神経をピリピリッとさせておかなくちゃいけないですし，よく考えなくちゃいけないし。それから，とても長い時間かかります。

なので，『バンゲリング ベイ』は，レベルマを1面クリアしただけです。これだけなのに，すっごく疲(つか)れちゃいましたぁ。

●ゲームの内容

『バンゲリング ベイ』は，バンゲリング帝国(ていこく)の島々に隠(かく)された「秘密(ひみつ)工場」を見つけ出して，全部壊(こわ)しちゃうゲーム，です。秘密工場を守るために，戦車，高射砲(こうしゃほう)，レーダーが，配備(はいび)されてます。そして，ヘリコプターを攻撃(こうげき)する戦闘機(せんとうき)，空母を攻撃(こうげき)する爆弾機(ばくだん)が飛んで来て，秘密工場を守ろうとガンバリます。

さて，『バンゲリング ベイ』で高得点(とくてん)を得(え)るには――

①マップを覚えます。

②遠い島から攻撃します。

③空母の近くのレーダーは，なるべく消します。

④戦艦(せんかん)は，建造(けんぞう)中に見つけて壊します。

⑤爆撃機は，後ろについて攻撃します。

というふうにするとよいみたい，です。

まず，マップは必ず覚えた方がよい，と思います。特(とく)に，戦艦が建造されるドックの位置は，覚えておきましょう。そうしないと，〈WARNING〉のサインが出ても，どこへ行っていいのかわからなくなっちゃいますもん。

それから，空母の近くに「レーダー」があると，爆撃機に攻撃されやすくなります。近くのレーダーは，気がついたら消しておきましょうネ。

爆撃機は，空母の横にヘリコプターを止めて，現(あらわ)れたところをやっつける迎撃(げいげき)スタイルと，追いかけてやっつける追撃スタイルの2種類あります。初めは，迎撃スタイルで，ヘリコプターの操作に慣(な)れたら追撃スタイルに代えます。

うーん，誘導(ゆうどう)ミサイルの避(さ)け方は，ぐるぐると旋回(せんかい)するのがよいみたい……。誘導ミサイルの燃料がなくなるまで待ちましょう。

登場キャラクタ

◉ヘリコプター

16方向に飛べて，32段階の速度調整ができる最新型戦闘ヘリコプター。

ヘリコプターには，無制限に使えるバルカン砲と，最大9発まで積める爆弾が用意されてます。

5機まで，あります。

◉レーダー(100ポイント)

ヘリコプターと同じ画面にレーダーがあると，戦闘機を呼ぶ能力があります。空母の近くにあるレーダーは，空母に爆撃機を呼びます。

時間がたつにつれて，数が増えていきます。

◉戦闘機(200ポイント)

ヘリコプターが接近したり，レーダーに誘導されて，コースを変えミサイル攻撃や体当たりをしてきます。

集団で攻撃することも，あります。

◉爆撃機(200ポイント)

ヘリコプターは無視して，空母だけを攻撃します。

〈ALERT〉のサインが出たら，すぐに空母に戻って爆撃機を倒しましょう。

◉高射砲(100ポイント)

ヘリコプターに向けて12方向に弾を撃ちます。

帝国に危機が迫まると，誘導ミサイルを撃ってきます。注意しましょう。

◉タンク(50ポイント)

工場付近を，ただ動きまわるだけのキャラクタ。ヘリコプターに，軽いダメージを与えます。

スペシャルを出す時，じゃまになります。

◉ボート(50ポイント)

海上を移動する，水陸両用の運搬車。高射砲，タンクなどを運搬しています。

ヘリに攻撃してきますが，ダメージは少ないです。

◉ミサイル

戦闘機から発射される直進型と，高射砲や戦艦から発射される誘導型の2種類があります。

誘導型は，逃げきって燃料が切れるのを待ちます。スピードは，ヘリコプターよりも速いので，急旋回して逃げます。

◉空母

ヘリコプターの修理と爆弾の補給します。一定航路を進めます。ヘリから見た空母の位置は，矢印でいつも指しています。

〈ALERT〉のサインが出ると，爆撃機の攻撃が始まります。一定のダメージで沈没してしまいます。

◉工場(5000ポイント)

最終兵器を開発しているバンゲリング帝国の工場。

全部で6か所にあり，高射砲，レーダー，戦闘機などを，製造しています。

すべての工場を破壊すると〈COMPLETE〉と画面に表示されて，次の面に進みます。

◉戦艦(5000ポイント)

バンゲリング湾内に1か所だけあるドックで建造される最後の敵です。

〈WARNING〉が画面に表示されると，戦艦が建造され始めます。2度めが表示されると，出航します。

1度めの警告が表示されたら，すぐに建造中の戦艦を見つけて破壊しましょう。5発くらいの爆弾で，破壊できます。

工場を撃破

工場を見つけたら，まず周りのレーダーや高射砲を消しましょう。

工場の周りに敵がいなくなったら，工場を攻撃します。工場の上に止まって，爆弾を落とします。

工場は，時間がたつとだんだん修理されちゃうので，同じ工場を破壊するまで攻撃しましょうネ。工場は，遠くのから破壊する方が，よいみたい。

↓

空母を爆撃機から守る

〈ALERT〉は，空母が爆撃機から攻撃されてることを知らせるメッセージ。この〈ALERT〉が表示されたら，どんな時でも，すぐに空母に飛んで帰りましょう。

空母が沈没すると，ヘリコプターの修理ができなくなります！

だいたい2，3機の爆撃機が襲ってきます。

爆撃機を倒すには，後ろにまわって攻撃するのが一番。でも，空母の近くで静止して，迎え撃つのも悪くないです。

敵の飛行機で補給する

ヘリコプターは，爆弾を9発までしか積めません。なので，空母と工場を行ったり来たり…。

こういう爆弾の補給だけの時，敵の飛行場を利用しましょう。ただ，とても危険なので，注意，注意。

↓

↓

↓

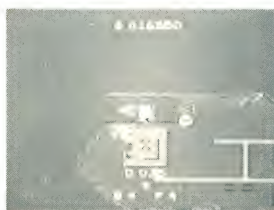
戦艦を撃破！

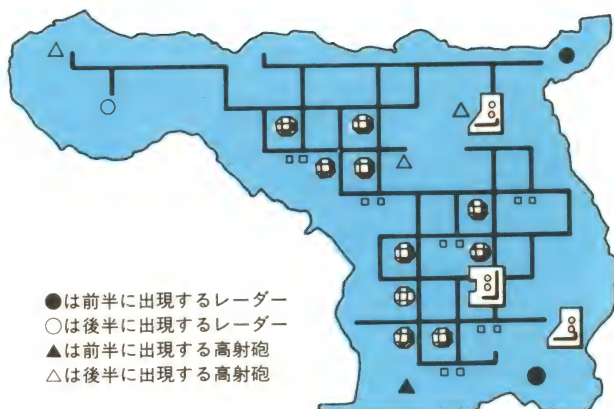
1回目の〈WARNING〉が表示されたら，すぐに建造中の戦艦を見つけて破壊しましょう。戦艦が出航したら，まず無理です。

すぐにドックへ直行できるように，ドックの位置を調べておくと，便利です。

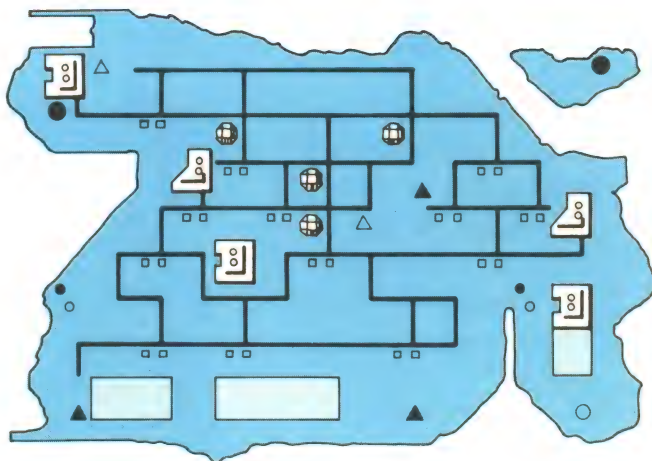
↓

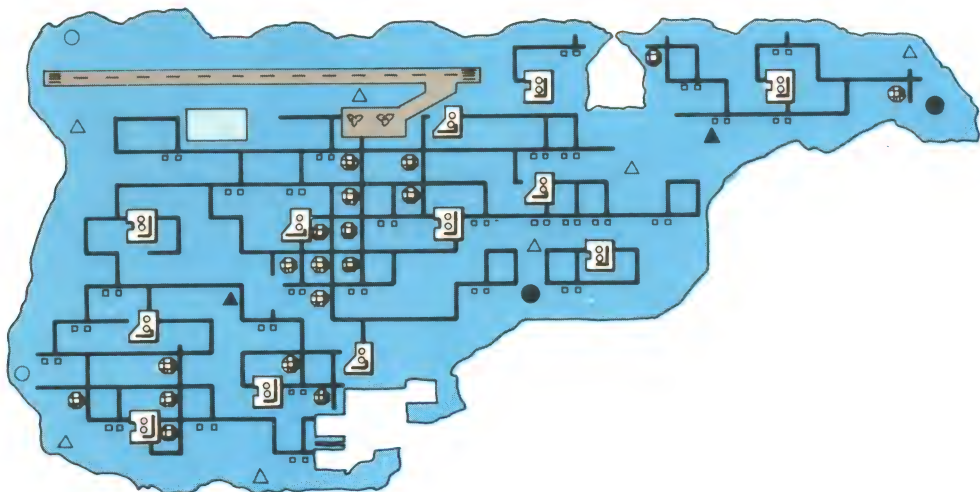
↓

↓

第1目標

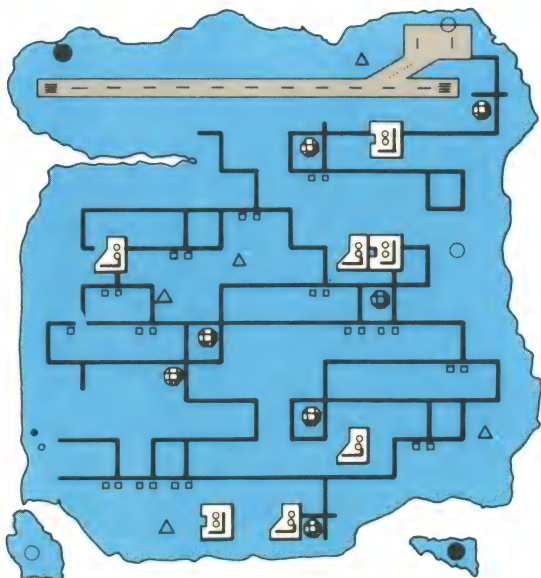
「バンゲリング帝国(ていこく)マップ」の中央に位置する島です。この島は，空母から一番遠い工場のある島です。後々のことを考えて，この島から攻撃しましょう。工場は7～9発で破壊(はかい)できると，思います。

第2目標

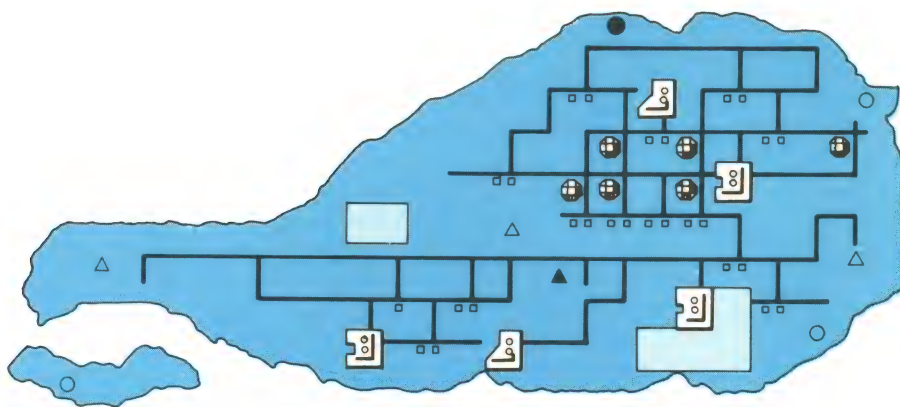
一見ふつうの島。でも，島の左上に戦艦(せんかん)を建造(けんぞう)するドックがあります。

〈WARNING〉の時，直行できるように，島の位置をキチンと覚えておくと便利です。工場は，だいたい9発くらいで破壊できると，思います。

第3目標

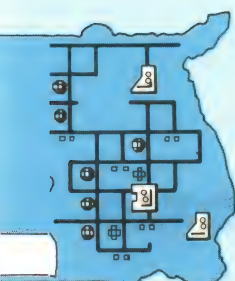
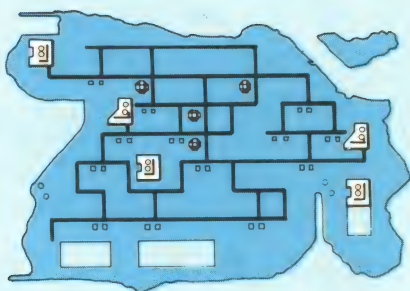
バンゲリング帝国の中でも，一番大きな島です。大きいだけじゃありません。戦略上でも，一番大切な島です。島の上部には，飛行場があります。この飛行場にヘリコプターをとめて，爆弾を補給できます。上手に，利用したいですネ。工場は，２か所あります。工場を破壊したあと，残ったレーダー，高射砲はキレイに片づけておきましょう。

第4目標

第３目標の島と比べると小さいけど，なかなかの大きさの島。

この島にも，飛行場があります。でも，この飛行場にヘリコプターを止めることはできませんよお。工場は，２つだけです。

第5目標

棒のように長い島と，小さな2つの島からできてる島です。島の中央に，工場があります。工場を攻撃(こうげき)する前に，高射砲(こうしゃほう)，戦車，レーダーを消しておけば，難(むずか)しくなかったんです。

バンゲリング ベイ

ゲームマップ

けっきょく南極大冒険

『けっきょく南極大冒険』は，ペンギン君がすっごくカワイくって，なかなか気に入ってます。こういうゲーム，とっても好きですネ。特に，ペンギン君が，オットット，とつまずくとこが好き。

でも，ゲームがちょっぴり簡単すぎる，と思いました。

もう少し複雑なゲームの方が，遊びがいがある，みたい…。結論は，『けっきょくジャングル大冒険』がよいと思います。

●ゲームの内容

『けっきょく南極大冒険』は，コントローラーとジャンプボタンを上手に操作して，ペンギン君を「日本基地」まで連れてってあげるゲーム，です。氷穴やクレバスは，ジャンプして跳び越えます。氷穴からは，たまにアザラシが出てきます。よけてくださいネ。旗や魚は取ると得点になります。タイマーが0になる前に，目的地に着かないとゲームオーバーになります。

この『けっきょく南極大冒険』で，高得点を取るには——

①ペギコプターを，なるべく手に入れます。
②アザラシは，左右に移動しながらジャンプ。アザラシの横を通り抜けます。
③連結しているクレバスや氷穴は，一気に飛びます。
④スピードは，最高速で。
の4つを守るようにすると，よいみたい，です。

イー アル カンフー

『イー　アル　カンフー』は最初に見た時，とっても難しいゲーム，だと思いました。
だってコントローラを操作して出した9つの技で，闘わなくちゃいけないんだもん。わたしには，ちょっと無理です。
でも，ホントは，そんなに難しくなかったんです。
慣れると，何面でも消せるようになりました。

●ゲームの内容

『イー　アル　カンフー』は，コントローラとパンチ，キックボタンを操作して，次々に来る敵を倒すゲームです。敵の技や武器をかわして，必殺技を決め相手のエネルギーを0にします。敵のエネルギーが0になると，敵は倒れてしまいます。でも，自分のエネルギーが0になると，反対に自分が倒されてしまいます。
それで，『イー　アル　カンフー』で，高得点を取るには——
①相手との距離を，上手にとります。
②飛びげりで，決めます。
という2つで十分です。
敵には，棒術使いの「王（ワン）」，火炎術師の「挑（タオ）」，くさり使いの「陳（チン）」，シュリケン使いの「藍（ラン）」，謎の男「呉（ウー）」の5人がいます。
「棒術使いのワン」を倒すには，棒にやられないくらい近づいて，飛びげりで決めます。パーフェクトは，簡単に出せます。
2人目の「火炎術師のタオ」は，敵から技をしかけられない程度に近づいて，必殺ワザで決めます。ワザは，飛びげり，ハイキック，足払いを使います。全部飛びげりでも，大丈夫です。
3人目の「くさり使いのチン」の時は，まず十分に近づきます。そうすると，くさりを投げてきません。あとは，必殺ワザで決めます。ハイキック，足払いを上手に使い，飛びげりで決めましょう。
4人目の「シュリケン使いのラン」の時は，ぎりぎりまで近づきます。ちょっと離れちゃうと，すぐにシュリケンを投げてきます。
5人目の「謎の男ウー」の時も同じです。ただ，ウーはとても足技が上手なので，飛びげりで決めましょう。
あとは，また1人目に戻ります。
謎の男ウーの技は，体当たりです。体当たりでぶつかってくるので，ジャンプして避けましょう。

V3.0について

　この本では，『ファミリーベーシックV2.0』を中心に書かれています。
　けれども，『ファミリーベーシックV3.0』についても，一応，説明しておきたいと思います。
　ファミリーベーシックV3.0では，次のような機能が強化されています。

①メモリ容量が2倍に，なった。
　V3.0では，ユーザが使えるRAM容量を，V2.0の2倍の4Kバイトに拡張してあります。なので，より多くのプログラムを作ることが，できます。
②プログラムを作成する時に便利な命令が，増えた。
・AUTO（オート）→行番号が自動的に付く。
・RENUM（リナンバー）→行番号を変更する。
・DELETE（デリート）→複数の行を削除できる。
・FIND（ファインド）→変更したい行を自動的に探す。
③プログラムのエラーチェックのための命令が，用意された。
・TRON（トレースオン）
・TROFF（トレースオフ）　→実行状況を調べる。
・ON ERROR GOTO
・RESUME
・ERR　→エラー処理を行う
・ERL
④BG面をバック・アップできる。
⑤etc.

テータイム

ふんわり、フワッフワ、背のびしたいナ

「こどもの国」へ行ってみたり、ちょっとした「思い出」を、書いてみたりしました。なんとなく、書いてしまったんです。なので、「なんで、こんなページがあるんだ？」と思う人も、いるかもしれませんね。まぁ、あまり気にせず　サラッと読み流してくださいませ。ようするに、息抜きをするページ、です。わたしたち２人も、十分に息抜きをしちゃいました。なお、このページは斉藤千秋、木村香奈枝の２人が担当しました。

●それぞれのプロフィール

▶斉藤　千秋（さいとう　ちあき）

・生年月日……昭和45年　**4月30日**です。
・血液型……明るくって，楽天的な**O型**！
・星座……**おうし座**
・スキな食べもの……色々ありすぎて，こまっちゃいますねー。最近は，**おしんこ，**かな。
・キライな食べもの……きらいな食べ物って，ちょっと思いつかないです。なんでも，おいしいっ！
・得意なこと……**バトミントン**。学校では，バトミントン部に入ってます。いちおう，選手だったりします。得意！　なんて言えないけど，あみ物，好きですねー。
・スキな飲みもの……**ミルク**。朝，起きた時，コクコクコクッ，飲みます。

▶木村　香奈枝（きむら　かなえ）

・生年月日……昭和45年　**6月18日**。
・血液型……ほんとうは，明るくって，楽天的な**O型**，なんです。でも，A型でしょ，とよく言われます。
・星座……**ふたご座**
・スキな食べもの……**サラダ**。サラダに限らず，さっぱりした食べものが，好きみたい，です。
・キライな食べもの……**お肉**。とくに，牛肉に弱いんです。なぜでしょう？
・得意なこと……うーん。得意，なんて言えるのって，ないんです。ただ，**フルート**が，ちょっと好き…。
・スキな飲みもの……**オレンジ・ジュース**。

●香奈枝の思い出――1

修学旅行。

わたしたちの学校は，「京都と奈良」へ行きました。

2泊3日，です。

東京の学校は　普通，「京都と奈良」へ行くみたい…。うーん，なかには「北海道」とか「九州，広島」へ行く学校も，あるらしいですね。わたしは，お友だちと一緒なら，どこでもイイナ。

修学旅行の思い出，とってもたくさん。

その中でも，いちばん，心に残っているのは，正座したまま眠っちゃったこと，です。

1日目は，ホテルに泊まりました。

ホテルと言っても，そんなにリッパじゃなくて，普通のホテルでした。あっ，こんなこと言って，ごめんなさいっ。泊めてくださったのに…。

でも，ホテルのことは，ぜんぜん気になりませんでした。お友だち全員，同じ所に泊まってる。それだけで，うれしくって…。

クラスの女の子どうし，男の子どうしが，6，7人ずつ1つの部屋に寝ることになりました。

寝るのは，7時半。

でも。

今日は，修学旅行の1日目，です。女の子が7人，です。

これじゃあ，どんなに「よい女のコ」でも，すぐ眠っちゃダメですよ，て言われるのと同じ，と思います。すぐ，おふとんにもぐって，おやすみなさいクークー，とはいきません…。

●香奈枝の思い出——2

初めは，おふとんの中に入って，おしゃべり，しました。

女子はお菓子を食べたり…。本当は，お菓子を食べたりするのは，いけません。

でも，これだけなら，よかったんです。

隣の部屋へ行って，まくら投げしましょ！　となってしまいました。よく考えると，よく考えなくってもそうなんですが，みんなが，しーんと寝てる時に，まくら投げ。先生に見つかっちゃわないほうが，おかしい，です。でも，その時は，「キャーキャー，はやく行こう，はやくぅ」。

それで，先生に見つかってしまいました…。

あたりまえ，ですね。

ドタドタ，バタバタ。それに，キャーキャー，ワーワー。もう，大騒ぎ。もう，とっても楽しかった，です。

すぐに，先生が飛んできて。「部屋へ戻って，朝まで正座してろ!!」と怒られてしまいました。

７人で，部屋で正座…。一人で怒られるのって，とってもつらい，です。でも，お友だちと一緒だったら，平気。不思議，ですね。

わたし，すぐに眠っちゃいます。ベットにもぐると，コテッ，すぐに眠ってしまいます。

なので，正座して，目を閉じて。目を開けたら，朝，でした。

本当は，わたしが眠ったあと。先生が「眠い者は，足をくずして眠ってもいいゾ」と言っていた，そうです。なので，わたし以外の子は，みんな，コロッとねっころがって眠ってました。

わたしは，そんなこと知らずに，ぐっすりと…。

朝，目が覚めてから，ちょっとたいへん，でした。もう，足が，ピリピリ，ピリピリ，シビレっぱなしで…。30分ぐらい，歩けませんでした。

忘れたい思い出は。

わたしが，本当は「ジュースの缶恐怖性」なのを，クラスのみんなに知られちゃったのが，なるべく忘れたい思い出，です。

えと，「ジュースの缶恐怖性」と言っても，よくわからない，と思います。これは，わたしが勝手に付けた名前，なんです。

ジュース恐怖性は，ジュースの缶を見ると，「こ，こわい！」とにげ出しちゃう，わけでは，ありません。ジュースの缶を見ても，大丈夫。

でも，ジュースの缶のフタ。

これが，問題なんです，わたしには…。

じつは，わたしは，ジュースの缶のフタを開けることができないんです。

まだ，一度も開けたこと，ありません。生まれてから14年間もっ！

いつも周りにいる人に，開けてもらっています。ちーちゃんみたいなお友だちなら，お願いして，開けてもらいます。でも，よく知らない人の時，困ります。「これ，開けてください」なんて，ちょっぴりはずかしい，です…。それで！　いつも，わたしのジュースの缶は閉じたまま。

これは，ずーっと，秘密，でした。

ちーちゃんみたいに，とても親しいお友だち，しか知らなかったんですよぉ。これも，修学旅行のおかげ。ううん，人のせいにしては，いけませんね。自分が，もう少し…。

今年の夏の宿題は，ジュースの缶を一人で開けられる，ことです。

●千秋の思い出──1

体育祭でのことが，いちばんの思い出，ですう。

体育祭では，わたし，2人3脚をすることになりました。体育で得意なのは，ぴょんぴょんとび箱とか，ころころマット運動。走るのは，あんまり得意じゃないんです。

それで，練習することにしました。

一緒に2人3脚するお友だちと，体育祭の前の日，練習したんです。近くの公園で，ちゃあんと体育祭とおんなじに体操着を着て。2人の足をしばって，公園の中を，いちに，いちに，走って練習しました。

初めは，30分くらいのつもり，だったのに。

なんだか，走るのが楽しくなっちゃって，2人で1時間くらい，走りまわってました。だって，2人の足がピッタリ動くんだもん。「これなら，ぜったい一等賞まちがいなしだね一」て2人で喜んでいました。

次の日の，体育祭。

2人3脚は，おべんと食べて，すぐに始まりました。

2人1チーム，6チームで競います。

1年生が終わって。2年生が終わって。2年生が終わって。

そして，3年生のわたしたちの番，です。「パーン!!」とピストルの音が鳴って…。

一生懸命，走りました。一生懸命，走りました。

でも，一生懸命すぎちゃった，みたい。

わたしが，転んじゃいました。他のチームは，どんどん，前へ走って行きます…。

なみだが，ポロポロ，こぼれてきました。
べつに，かなしくなかったのに。へんですネ。
なんだか，時間がストップした気持ちで…。2人で練習したこと。クラスで，一等賞取るんだあ，と喜んでたこと。おべんとのこと。前を走ってく人のこと。それが，ふわふわ風船みたいに，飛んでいる感じ，でした。
ヒックヒック，泣いてるわたしを，どうにかゴールまで，引っぱってくれました。
6位，でした。
こんな思い出って，ほんとは，忘れたい思い出，のはずなのに。なぜだか，すてきな思い出，になってます。
どうして，かなあ。

●千秋の思い出——2

忘れたい思い出って，よく考えると，あんまりないんです。

わたしって，忘れたいなあ，と思ってるうちに，ほんとに忘れちゃいます。

だから，ちょっぴり忘れたい思い出，にしますね。

冬服から，夏服へ，衣がえの頃です。

わたしたちの学校，冬服は紺のブレザー，夏服は，灰色のブレザー，になってます。

朝，今年はじめての，夏服，を着ました。

きのう，電話で，かなえちゃんと「あしたから，夏服を着なくっちゃあ…。うんうん，そう」なんて，話しました。

この朝は，遅刻，ぎりぎり。たいへん，たいへん。かなえちゃんと一緒に，すこし駆けていきました。

そうすると，まだ冬服のままの生徒がいました。

「あれ？　あの男の子，おかしいよぉ。まだ，冬服のまんま」と，わたし。

「ああほんと。まちがえちゃったのネ。」と，かなえちゃん。

でもぉ。

冬服の生徒が，何人もいると…。

ふんわり、背のびしたいナ。

朝、雨が降ってたんです。もう、心配しました。でもでも、それから一粒も降らなかったんですよ。神様を、信じてしまいます。

●かなえの感想文

お天気が、一番気になりました。
朝、パラパラ、雨が降っちゃうし…。天気予報では、くもり時々あめ、だったんです。
でも、思いきって、『こどもの国』へ行くことに決めてしまいました。
だって。ちーちゃんと2人だけで、どこかへ出かけるの、はじめて…。こんなに、なかよしなのに…ね。
けれども。『こどもの国』に着いてからは、雨、ぜんぜん降らなかったんです。時々、お日様も出てきました。
うーん、神様を信じたくなっちゃいます。

●ちあきの感想文

雨が降ったあとの景色って、すてきです。
アスファルトも、芝生も、ベンチも、周りの木々も、なにもかも、シャワーできれいに洗っちゃったみたい！
サラダみたいに、ひんやりシャキシャキ!!

わたしたちは、なぜだか、いつも一緒なんです。
学校でも、遊ぶときでも。性格も、好きなことも、
違うみたいなのに。磁石のNとS、みたいですね。

14才って。"夢"と"現実"との距離、まだまだ、だいぶあるみたい、です。
いつも、背のび背のび！
でも。キレイも、ステキも、キュートも、ポップも、イロッぽいも、欲しくないです。ただ、ひとつだけ、"かわいい女の子"のまま……。
だから──。
ふんわり、ふわっふわ、背のびしたいナ。

第2章

ファミコン・マイコンかんぜん基礎知識

この章では、マイコンと『ファミリーコンピュータ』の基礎知識について、説明します。プログラムを作れるようになるには、ちゃんとしたマイコンについての知識を知っておかないと、難しいと思います。やっぱり、後々、困ってしまいます。たとえば、インタプリタ、メモリ、LSI、CPU、ROM、RAM。知らないと、タイヘンです。でも、この章を読めば、大丈夫です。なお、この章は、木村香奈枝が担当しました。

こんなこと、説明します。

●ファミリーコンピュータ

『ファミリーコンピュータ*』で，遊んでいて。

ふっと。わたしは，こんなふうに思います。「うーん　ファミコンは　**どんな仕組みになってるの？**」，なんて…。

みなさんは，どうです？

わたしは，あんまり，ゲームで遊ばない方です（と，自分では思っています)。ゲームで遊ぶよりも，ゲームを作ったりする方が，好き，なんです。

でも，「宿題を，しなくっちゃな！」なんていう時は，どうしても，ついつい，遊んでしまいます。

うーん，わたしって，ちょっぴり「意志(いし)の弱い子」のようですね。

こんなふうに，遊んでいて，思うんです。

どんな仕組み？

なんて――。

たぶん，みなさんも　わたしと同じように思ったことあるのではないでしょうか？

ほんと，**不思議(ふしぎ)**，です。

けれども，その「不思議な疑問(ぎもん)」の答えは，誰(だれ)も，教えてくれません。もう，困(こま)ってしまいます。

そこで，この「2章」では

	内　容
1	ファミリーコンピュータの中身
2	ファミリーコンピュータの仕組み
3	BASIC
4	プログラム
5	etc.

について，説明できたらいいなあ，なんて思っています。

ファミコンの中身は…。

まず，初めに『ファミリーコンピュータ』の

中身を見てみたい

と，思います。

いつも，外側ばかり見ているんですもの。たまには『ファミリーコンピュータ』の，中身を見てみたいな。

どうです？

みなさんは，わたしみたいに，「中身をのぞいてみたいな」なんて，思いませんか。

わたしは，もう，すぐに思ってしまいます。

目ざまし時計に，テープレコーダー。マイコンに，スピーカー。

何でも，中身を見れたらいいなー，と思います。

でも，わたしは，思うだけなんです。

思うだけで，実際(じっさい)にばらばらにすることは，できません。

やっぱり，ちょっと，怖(こわ)い。

「電気」が入っているんですもの，電気製品(せいひん)には。ばらばらにしていて，もしも **「ビリッビリッ！」** なんてなったら…。

想像(そうぞう)しただけで，怖い感じします。

それに，ばらばらにしたら，**元に戻(もど)す自信、ないんです。**

元に戻せない自信なら，もう，たくさん用意してあるんですけれど（もし，欲(ほ)しい人がいたら，あげます）。

●ハード
ハードウェア（hardware）のこと。金物。
計算機を作っている物理的な要素。プログラムに対立する装置のこと。

ハードウェア
↕
ソフトウェア

「ハード」という物は，人が，直接見たり，さわったりできる物です。
例えば，えんぴつやはさみは，直接見たり，さわれますね。
えんぴつやはさみは，ハードなんです。
で，ここでは『ファミリーコンピュータ』の機械の部分のことをいいます。

分解してみました

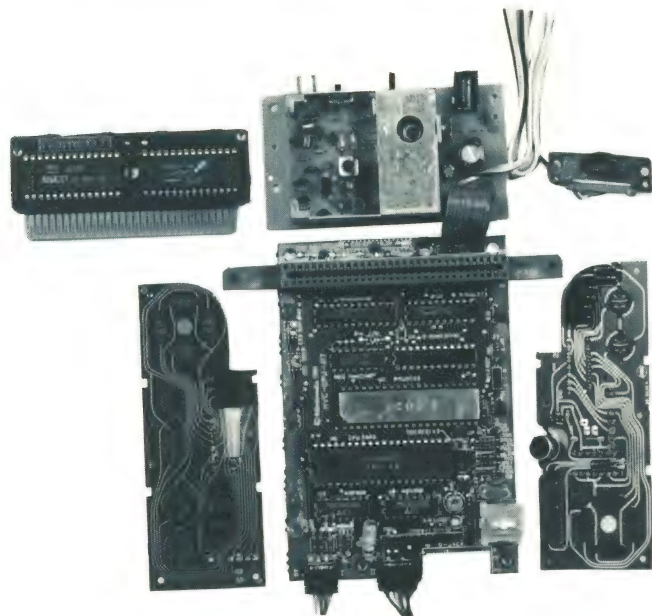
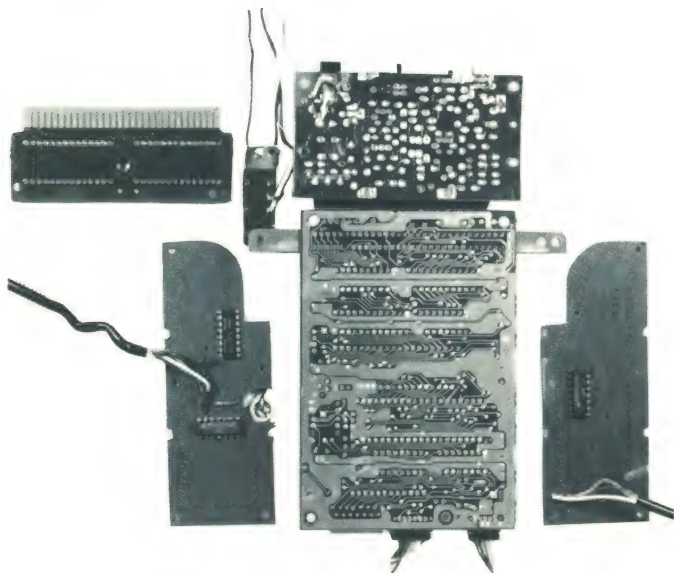
では，分解(ぶんかい)してみたいと思います。

でも，わたしは，苦手です。そこで，**「ハード*」の分かるスタッフの方**に，頼(たの)みました。

やっぱり，慣(な)れている人は，違(ちが)いますね。

おしゃべりしながら，どんどん分解してしまいました。

次の２ページは，その**分解した「写真」**です。

●ファミリーコンピュータを分解した写真(オモテ)

●(ウラ)

ICは，これらのトランジスタなどいろいろが，組み合わさって

計算したり，数値を記憶したり

しています。

このICは，次のような電気製品に，利用されてます。

製品名	ICの主な仕事
テレビ	チャンネルの選択
計算機	数値計算
デジタル時計	時間計算，時間・日付記憶
ビデオ	チャンネルの選択，微調整

他には，

使われる度合	製品名
◎	マイコン
△	自家用車
○	電子レンジ
○	ステレオ
△	ミシン
○	エアコン
○	電話
◎	電子体温計
◎	CD，VD*

なんていう製品に，使われていることもあるそうです。

自動車やミシンにまで，使われることあるなんて。ちょっとした驚きが，ありました。自動車なんて，どこに使われてるんでしょ？

うーん，電気に関係のある物，何にでも，使われているんですね。

「IC」は電気製品の

頭脳の役目をしてくれる

んです。

● CD，VD

CD（コンパクトディスク）とは，レーザー光線によりディスク面から音を再生できるようになっている円盤状の物。

ほこりや傷などの影響を受けにくく，音質が今までのレコードと比較し，優れている。

VD（ビデオ・ディスク）とは，ビデオ画像が記憶されている円盤状の物のこと。VDプレーヤにより再生する。

現在，LD（レーザービジョンディスク）方式とVHD方式の二種類が市場に存在している。

● LSI（エルエスアイ）
大規模集積回路のこと。チップ当たりの素子数が1000個～10万程度のものを指す。

● CPU（シーピーユー）
中央処理装置のこと。デジタル計算機の神経中枢であり，他のすべての装置の動作を統合，制御し，またデータに適用されるすべての算術または論理演算を実行する。

● ROM（ロム）
読出し専用メモリ。

● RAM（ラム）
読み／書き可能メモリ。

例えば，テレビ。
昔(むかし)なら，手でカチャカチャ，チャンネルを回します。
でも，今は，リモコンで，ピッ！
これは，ICが人間の代わりにチャンネルを回してくれるから，なんです。人が**「考える」**代わりを，ICがしてくれています。
それから　デジタル時計。
最近，電話番号を憶(おぼ)えてくれるデジタル時計があるのを，知っています？　とっても便利だそうです。
あれは，人が**「憶える」**代わりを，ICがしてくれています。

CPUとRAMとROM

ICには，いろいろな種類があります。
例えば——

名称(めいしょう)	正式名称
LSI*	Large Scale Integration
CPU*	Central Processing Unit
ROM*	Read Only Memory
RAM*	Random Access Memory

とか。
わたしは　こういう——**LSI、ROM、CPU**——というのは，苦手です。アルファベット3文字なんて，分かりにくいし，憶えにくいですものね。
みなさんも　こんな「アルファベット3文字」の名前，見たことありませんか。
そういえば。
『ファミリーコンピュータ』の

取扱説明書の16ページ

に，**LSI, CPU, RAM, ROM,** という名前が載(の)っていました。
みなさんも，一応，見てみてください。
こんなふうに，載っています。

そういえば，

前のページの図に，みなさんが初めて見る「言葉」がありませんでした？

例えば——

名称	正式名称
CPU	central processing unit
CRTC	CRT controler
ROM	read only memory
RAM	random access memory
TTL	transistor-transistor logic

なんて，難しそうです。どれも，アルファベット３，４文字のものばかりで…。

やっぱり，知らない言葉，不安になります。

なんとなく，気になって。特に，ベッドにもぐってから知らない言葉を思い出すと，**胸のあたりが，ムズムズしてきます。**

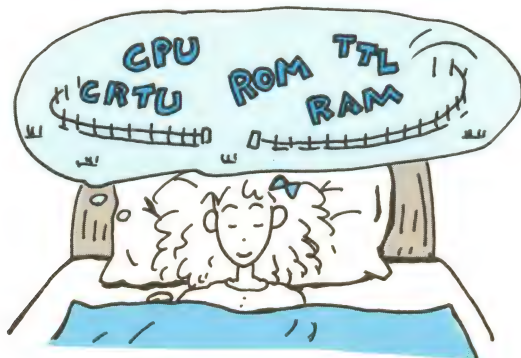
そこで，『ファミリーコンピュータ』に

使われている部品の説明

をしたいと思います。

これで，みなさんも安心して，ぐっすり眠れるといいですね。

こんな仕組みで動いています

もう一度，『ファミリーコンピュータ』の**内部の図**を，見てください。

● FM-NEW7

富士通が発表した，低価格高性能8ビットマイコン。コストパフォーマンスに優れている。

FM-7の血を受け継いでいる。

●カスタム

カスタムIC(custom IC)のこと。

注文された回路に基づいて作られたIC。

通常のマイコンでは，一般に流通されているICを利用することが多い。これは価格，信頼性，流通性が安定しているためである。

ただし，流通されているICを利用して，一定の回路を作ろうとした場合，どうしても数多くのICを組み合わせる必要がある。

ところが，カスタムICを使用すれば，注文された回路に基づいて作られるため，数少ない部品ですむ。

『ファミリーコンピュータ』の基板*の上に，〈左図〉のような物が，のってます。

真っ黒で，四角くて。それに，**なんとなく足，がはえている感じで…。**

これいったい何だと思います？

これは

チップ*

という物なんだそうです。

英和辞典で，調べてみると

chip　〈名〉
1　（木や石の）切れ端(はし)
2　（瀬戸物(せともの)の）かけたところ。かけら
3　（食物の）小片(しょうへん)

と，載(の)ってます。

「ポテトチップ」 と一緒(いっしょ)の，チップ。見ためは，チップとポテトチップでは　ぜんぜん違(ちが)います。でも　元の意味が一緒なんて，面白いですね。

でも，ボーイさんに渡(わた)す「チップ」とは，違います。

tip　〈名〉
1　チップ，祝儀(しゅうぎ)，心付け
2　軽く，たたくこと
3　（内々の）情報(じょうほう)　耳打ち

話が，ずれちゃいました。ごめんなさい。

それでも，少し，お役に立ったと思います…。

この四角くて，黒い「チップ」のことを

石

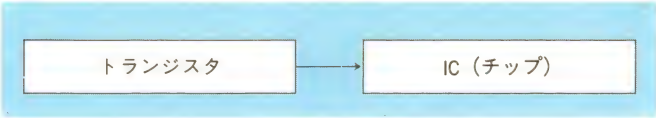
と，呼(よ)んだりもします。

ちょっぴり変な感じがします。**「石」** なんて。

●基板
プリント基板，プリント配線とも言う。
電気配線が絶縁物上にあたかも印刷されたかのような外観を備えたもの。

●チップ（chip）
数ミリ角のシリコン単結晶片に多数のトランジスタ，抵抗などの回路素子を作ったもの。
飛躍的な小型化，低価格化に，高信頼性が得られるようになった。

ようかんとか，チョコレートなら，よく分かります。形が，そっくり。でも，石，なんて…。

これは，昔むかし（十年くらい前だそうです）に，トランジスタ*のことを，**石**，と呼んでいたんだそうです。その習慣が，今でも続いているんですね。

そういう人は『ファミリーコンピュータ』の中身を，見て

「おおっ，石が少ないゾ！」

なんて，言うのでしょうね，たぶん。

でも，知らない人が聞いたら，もっと不思議に思いますね。

「えッ，ファミコンには，石が入っていたのか」なんて…。

想像しただけで、楽しくなります。

石が入っているんです。

「チップ」という名前は―あだ名―なんです。

本名は，日本語漢字四文字，

集積回路

という，名前です。「集積回路」だなんて，とっても偉い名前ですね。

日本語だけでは，ありません。

英語の本名もあります。

名称	意味
集積回路*	単一の半導体材料の上に，すべての部品が化学的に形成されている回路
IC	集積回路を，英語では IC（integrated circuit）と呼ぶ
チップ	集積回路の俗称

●トランジスタ
信号の増幅作用を持ち，従来の真空管に代わって使われる。
最近は数ミリ角のシリコンチップ上に数百〜数十万個のトランジスタ回路を作った集積回路（IC）として，数多く利用されている。

●集積回路
数ミリ角のシリコン上に，多数のトランジスタ，抵抗等の回路素子を作り，アルミ配線を行って作られる超小型電子回路。
パーソナルコンピュータ，デジタル時計，電子レンジ等，多くの電気製品に組みこまれている。

英語では

IC(アイ・シー)

と，呼ばれています。

このICは，Integrated Circuit，を略したものだそうです。

integrate	動 他
（各部分を全体に）統合する	
（部分を加えて）完全にする	
（温度・面積などの）総和を示す	
〔数学〕積分する。	

circuit	名
1 一周，巡回	
2 周囲	
3 巡回地域（round）	
4 〔電気〕回路，回線	

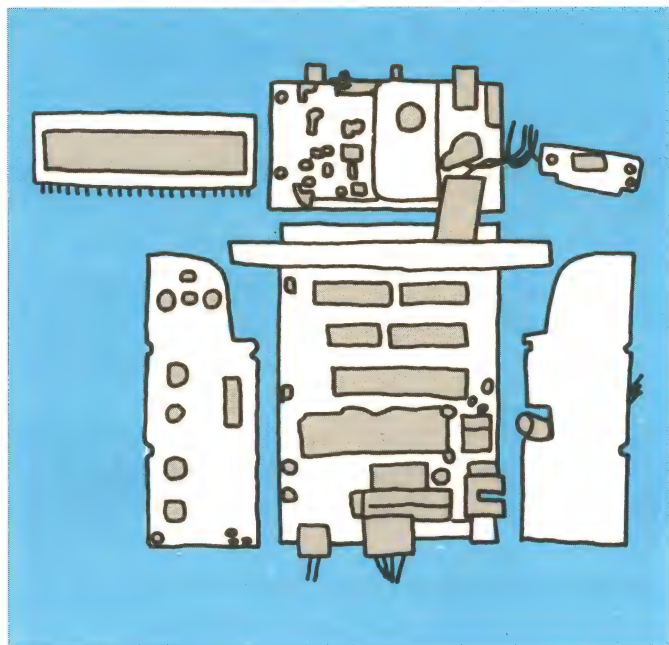
この「IC」の中には，とてもたくさんの **トランジスタ，抵抗，ダイオード** などが，入ってます。

だいたい，1000個から10万個，入っているそうです。

すごいんですね。

この数によって，ICは，四種類に分けられているそうです。

名称	意味
LSI	Large Scale Integration の略 大規模集積回路
MSI	Medium Scale Integration の略 中規模集積回路
SSI	Small Scale Integration の略 小規模集積回路
VLSI	超LSI

●ファミリーコンピュータを分解した図

●部品表

	部品の名
1	C P U
2	C R T C
3	R A M
4	T T L
5	電源
6	R O M(キャラクタ)
7	R O M(プログラム)

CPUとRAMとROM

『ファミリーコンピュータ』の，中身をのぞいた感想，どうでした。

「うーん，意外と部品数は少ないな」

「いやあ，思っていたより，ぜーんぜん多いですよぉ」

みなさんの感想は，どっち？

わたしは，『ファミリーコンピュータ』の中身を見て，

「ええっ、こんなに少ないんでしたの」

と，ちょっぴり驚いてしまいました。

普通のマイコン——例えば，日電*の **PC-8801MK II SR***、富士通の **FM-NEW7*** だったら，うーんと，多いはずです。

部品数が少ない理由を，スタッフの方に尋ねてみました。

「どうしてこんなに少ないのに，動くんですか」

「まず，ファミリーコンピュータの仕組みが，簡単である，ということ。そして，カスタム*を利用しているために，いっそう部品数が少なくてすむ」

ということでした。もう，ぜんぜん分かりませんよぉ。

「カスタム」って，何なんでしょうね。

こういう時は，国語辞典を引くのが，いちばん。

ちゃぁんと，載っていました。

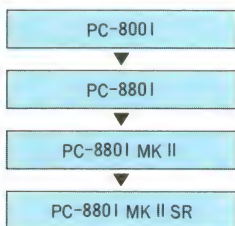
すごいですね。

カスタム	custom
①習慣，風習。 ②あつらえの。注文の。	

これは，②「あつらえの，注文の」だと思います。①だと，なんだか変です。まさか習慣や風習を利用すると，『ファミリーコンピュータ』の部品数が少なくなる，なんて思えないです。

たぶん，「注文の○○」のことを，カスタムって呼ぶんじゃないのかなあ，わたしの勝手な想像ですけど…。

●日電（にちでん）
マイコン界では，NECのことを，「日電」と呼ぶ習慣があります。
マイコン界七不思議の一つです。（香奈枝）

● PC-8801MK II SR
日電が発表した，高性能8ビットマイコン。
このクラスのマシーンでは，最高の人気，最高のソフトを誇る。
PC-8801，PC-8801MK II の血を受け継ぐ。

実力，人気共に，No.1のマイコンです。
なお，「PC-8801MK II SR」の「SR」は，意味なく付けたそうです。
「SR」が付いていると，カッコ良いですものね。
（香奈枝）

・形名　HVC—001

使用電源	専用 AC アダプタ
	DC　10V　850mA
消費電力	約 4 W
LSI	CPU　8 ビット　N—CH カスタム　1 個
CPU	(ビデオコントローラー)　N—CH カスタム　1 個
RAM	2 k× 8 ビット　スタティック RAM　2 個
ROM	最大32k× 8 ビット
本体寸法	幅150m/m　奥行220m/m　高さ60m/m
重量	約620g　(Ⅰ Ⅱ コントローラーを含む)
付属品	AC アダプタ (HVC—002)

RF　SWITCH (HVC—003)

75Ω／300Ω　変換器 (HVC—004)

ちゃんと，載ってます。

でも，**LSI**とか**CPU**とか，意味が分からないなんて…。みなさんだって，つまりませんね。

なので，この

名　称	正　式　名　称
LSI	Large　Scale　Integration
CPU	Central　Processing　Unit
ROM	Read　Only　Memory
RAM	Random　Access　Memory

という，難しそうな「言葉」の意味を，説明したいと思います。

LSI は学級委員長です

まず　上図の中から

LSI(エル・エス・アイ)

について，説明します。

この「LSI」，前に　チラッと，出てきました。憶えてます？

● 回路
普通は電気回路を意味し，導体などから作られた電流の通路をいう。

名称	意味
LSI	Large Scale Integration の略 大規模集積回路
MSI	Medium Scale Integration の略 中規模集積回路
SSI	Small Scale Integration の略 小規模集積回路
VLSI	超 LSI

こういうふうに，出ていたんです。
つまり，LSI という物は

大きな規模のIC

のこと。図をみれば，よく分かります。
でも，**「大きな規模」**というのは，IC の大きさ，じゃあないです。IC の中に入ってる，回路*の量の多さ，のことです。
うーんと，たくさんの回路が入っている方が，複雑な仕事ができます。

同じ一つの「IC」なら，複雑な仕事ができた方が，いいです。
どんどん大規模化して，最近では，超々LSI なんていう，すごいのまで作られています。
超々LSI，なんて。
名前だけで，ほんと，すごいことができそうです。

CPU*は

中央処理装置

のことです。

CPUは，主に，

	CPUの主な仕事内容
1	他のすべての装置の動作を管理する
2	プログラム*命令に従い算術または論理演算を行う

という仕事を，しています。

1	他のすべての装置の動作を管理する

C P U

装置A　装置B　装置C　装置D

考えたり，判断したり，計算したり…。
つまり，CPUは頭脳の働きをしてくれてます。
「学校」にたとえると，学級委員長。
クラスの意見をまとめたり，先生に相談したり。
そういう役目を，CPUはしています。

●CPU（シーピーユー）
中央処理装置ともいう。CPUは，デジタル計算機の神経中枢であり，他の装置の動作を統合，制御し，またデータに適用されるすべての算術または論理演算を実行する。

●プログラム（program）
電子計算機によって，所定の問題を解くために組み立てられた命令の集まり。

●メモリ（memory）
記憶装置ともいう。
コンピュータの内部記憶装置を指すのに，普通使われる。
ただし，厳密な意味では，CPUにより直接指定されるロケーションを意味する。
ディスク等の補助記憶装置は，含まない。
しかしながら，補助記憶装置もディスク・メモリというように呼ばれることもある。この時，内部記憶装置は，メイン・メモリと呼び，補助記憶装置と区別するのが一般的である。

メモリは憶(おぼ)えるIC

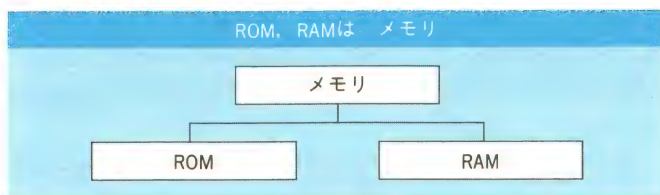
今度は **ROM** と **RAM** という言葉について，です。
ICの**種類**には，「CPU派(は)」と「メモリ派」の2種類があります。

CPUは「考える」IC。
でもメモリ*は

「憶える」IC

です。
何を憶えてくれるのか，というと

数　値

を，憶えてくれます。
「アリガトゴザイマシタ」なんて，おしゃべりする自動販売(はんばい)機や，電話番号を憶えていてくれるデジタル時計には，このメモリが使われています。
おしゃべり自動販売機は，声を，数値(すうち)になおして記憶。
デジタル時計は，名前や電話番号を，数値になおして記憶します。
そうそう，「ハッピーバースディ・トゥユー!!」と，音楽が鳴るバースディカード。あれにも，このメモリが使われています。
音符(おんぷ)を，きちんと憶えているのかな，と思います。
声を憶えたり，音符を憶えたり。メモリは，なかなかのがんばり屋さん，なんですね。
そして，ROMやRAMは，この**メモリの種類**なんです。

RAM*は
読み／書き可能メモリ

です。

RAMには，いろんな数値を，たくさん書き込めます。その書き込んだ数値を，読み出すこともできます。

ですから，電話番号を記憶しているデジタル時計は，RAMに，名前や電話番号を記憶していますね。

ただ，電源を切ると，それまで記憶していた数値は，ぜんぶ消えて無くなってしまいます。

せっかくいっしょうけんめい憶えたのに，消えて無くなっちゃうなんて。なんだか，かわいそう…。

う〜ん。でも「もう，**はずかしい!!**」って思い出も，パッと消えちゃいます。なかなか良いのかもしれませんね。

ROM*は
読み込み専用メモリ

です。

RAMは，自由に数値を書き込んだり，読み込んだりできました。

でも，ROMは違います。

ROMは　読み込むことしかできません。

ROMを作る時に，記憶してもらいたい数値を，書き込んでし

● RAM（ラム）

Random Access Memoryの略。読み／書き可能メモリともいう。

RAMは，通常，電源を切ると記憶内容が消えてしまう揮発性メモリ（volatile memory）である。

RAMには，スタティックRAM（static RAM）とダイナミックRAM（dynamic RAM）の二種がある。

スタティックRAMは，静的で安定した記憶方式である。ダイナミックRAMは，動的で不安定な記憶方式である。そのため，周期的に再度書き込みをする必要がある。

● ROM（ロム）

Read Only Memoryの略。読み込み専用メモリともいう。

ROMは，電源を切っても記憶内容が消えない不揮発性メモリ（non-volatile memory）である。

マイコンの固定したプログラムなどのように記憶内容を書き変えないメモリには，ROMが用いられる。

ROMには主に，マスクROM，PROM，EPROMの3種がある。

マスクROMは，製造時に記憶内容をも書き込んでしまい，永久に内容を書き変えることはできない。

PROM，EPROMは通常の動作では記憶内容を書き変えることはできない。

まいます。この時に記憶した数値は，書き変えることができないんです。電源を切っても，記憶した数値は消えないので，安心。

『ファミリーコンピュータ』のゲームは，このROMに記憶されています。

ですから，『バンゲリング・ベイ*』**や**『ギャラガ***』も，このROMに記憶されています。

電源を切っても，消えて無くなったりしないんですものね。

RAM，ROM，っていう2種類のメモリについて分かってもらえたと思います。

「よく分からなかったな」という人います？　ごめんなさい。わたし，もう少し上手に説明できるように，練習します。

でも，安心してくださいね。それほど，くわしく知ってる必要ありませんから。何となく，で十分なんです。

そうですね。

ROMは教科書，RAMはノート

という，感じ…。

しかし，特殊な方法により書き込むことが可能である。

●バンゲリング・ベイ

『ファミリ・コンピュータ』用ゲーム。発売元，株式会社ハドソン。

100画面分のスクロールとボイス機能をそなえた画期的なファミリーコンピュータ用ソフト。

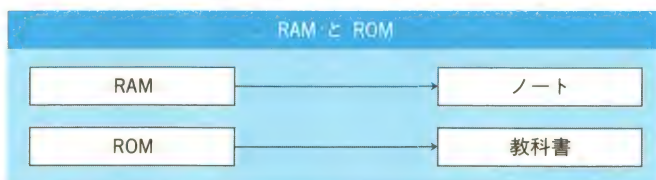
広大なバンゲリング湾に潜入し敵の秘密工場を破壊するリアルタイム・シミュレーションゲーム。

●ギャラガ

『ファミリーコンピュータ』用ゲーム。

ブームとなった「ギャラクシアン」を継承するゲームとして，1981年にナムコが開発。

エイリアンの動きやデュアルファイターなどゲーム内容が進歩し，根強い人気が続いている。

1985年には，ファミリーコンピュータ用カートリッジソフトとして発売される。

分類してみました。

『ファミリーコンピュータ』の部品の説明は，終わりました。
どうでした？
わたしの説明で分かってもらえたのなら，うれしいです。
『ファミリーコンピュータ』の中には，いろいろな部品が入っていました。

名称	意味
CPU*	中央処理装置
ROM*	読み込み専用メモリ
RAM*	読み／書き可能メモリ
TTL	トランジスタ　トランジスタ　ロジック
CRTC*	画面コントロール用 CPU

どれも，難しそうな，覚えにくそうな名前ばかり…。
うーん，『ファミリーコンピュータ』を動かすために，こんなにたくさんの部品が，活躍してたんですね。
ほんと，ごくろうさまです。
けれど。
こんなにたくさんの部品が，活躍してる

『ファミコン』は何者？

なのでしょう，いったい。

マイコン村の住人だそうです。

『ファミリーコンピュータ』を，動物みたいに

分　類

してみたいと，思います。
こうすれば，『ファミリーコンピュータ』の正体が，よく分かる

● CPU（シーピーユー）
Central Processing Unit の略。
中央処理装置ともいう。
CPU は，デジタル計算機の神経中枢であり，他のすべての装置の動作を統合，制御し，プログラム命令に適用されるすべての算術または論理演算を実行する。
マイコンにおいての CPU は，マイクロプロセッサと呼ばれる LSI チップである。

はず，です。

『ファミリーコンピュータ』は　分類してみると

電気製品目 マイコン科 テレビゲーム属 ファミリーコンピュータ種

ファミリーコンピュータを分類すると…

目	電気製品目
科	マイコン科
属	テレビゲーム属
種	ファミリーコンピュータ種

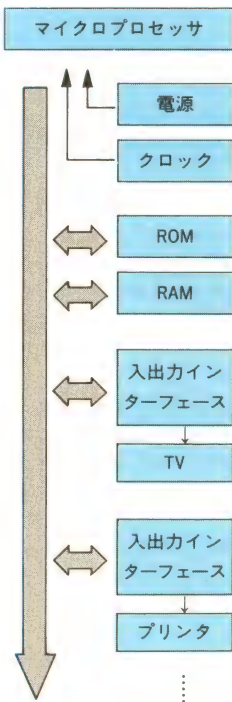
と，なるそうです。

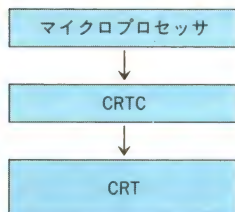
生き物でもないのに，分類しちゃう，なんて…。とっても不思議感覚。

この方法は，スタッフの方に教えてもらいました。

こういうように分類すると，その物がどういう物なのか，よく分るそうです。

確かに，そうですね。

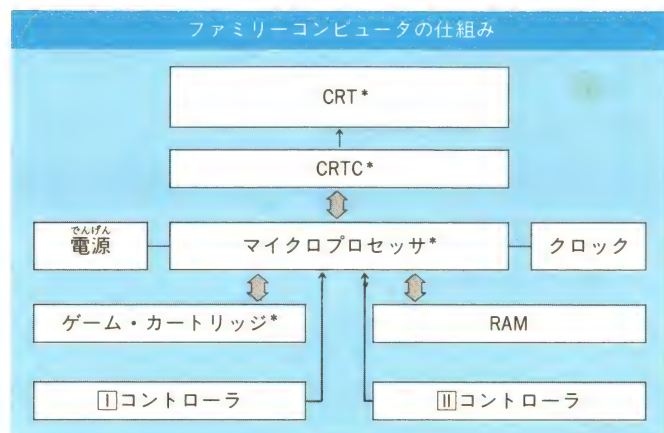
この分類図を見ると，『ファミリーコンピュータ』が，どこの誰なのか，すぐに分ってしまいます。

● CRT（シーアールティ）
Cathode Ray Tube の略。
陰極線管ともいう。
電子ビームが電子レンズにより制御される電子管。管の表面上に情報を表示したり，管表面の励起位置の形でデータを記憶できる。

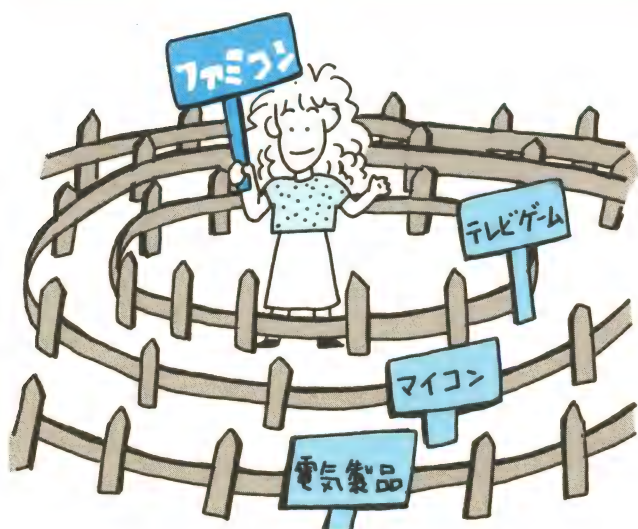
● CRTC（シーアール ティシー）
CRTcontroler の略。
CRT コントローラともいう。
CRT を制御するための CPU のことである。画面上に表示するキャラクタの管理，色設定，表示等を行ない，マイクロプロセッサを助ける。

● マイクロプロセッサ
マイコン（マイクロコンピュータ）は，通常のデジタル計算機と同様に，CPU，主記憶装置，制御装置，入出力装置等から構成されている。
マイクロプロセッサは，マイコンの CPU に相当する回路群であり，単一の LSI チップに収められている。「ファミリーコンピュータ」では，6502を改良したカスタムマイクロプロセッサを使用している。

● 6502
米社製8ビットマイクロプロセッサ。
コンパクトにまとめられた命令群を持ち，高速実行が

『ファミリーコンピュータ』は　電気製品町のマイコン村，テレビゲーム家のファミリーコンピュータさん，なんですね。
こういうの分りやすくて，わたし，好きです。

こんな仕組みです

そして『ファミリーコンピュータ』は　次の図のような仕組みで動いています。

特長である。
PET-2001，APPLE II 等の名機に使用された。

●ゲームカートリッジ
この図では「ファミリーコンピュータ」用ゲームカートリッジを指す。
カートリッジ内には，プログラム用 ROM16K バイト，キャラクタデザイン用 ROM16K バイトが装備されている。

BASICは、オムレツです。

● BASIC（ベーシック）
Beginner's All Purpose Symbolic Instruction の略。
デジタル計算機のコンパイラー言語の一種として開発された。現在，パソコン用プログラム言語として普及している。
命令は比較的短くて使いやすく，FORTRAN（フォートラン）などを学ぶ以前のもの。行列演算に便利な命令も用意され，文字の処理が可能な拡張 BASIC もある。

● プログラム言語
デジタル計算機のプログラムを書くために用いられる言語。
計算機を直接動作させる符号命令系である機械語（マシン語）から，人間の言語に近い自然語まで種々のものが開発されている。
現在は機械語やそれに近い言語で書くことは特定の人しか行わず，人間に分りやすく，計算機で共通性のあるような各種のプログラム言語がある。

みなさんは　たぶん

BASIC*(ベーシック)

という名前を，既に聞いたことがある，と思います。
例えば，『ファミリーベーシック』。
少しマイコンにくわしい人なら，PC 8801 シリーズや FM－7 に使われているのを，知っているかも。
『BASIC』は　普通の英語では

basic	名
1．基礎，基準，根拠	
2．（薬などの）主成分。	
or the basic of…を基礎として，…の基盤の上に	

という意味，だそうです。「基礎」，「基本」なんていう意味に，なっています。
でも　マイコンの世界では，べつ。
マイコンの世界では　『BASIC』は

「プログラム言語*」の名前

です。

プログラム言語は、目玉焼です

「プログラム言語」。
また，新しい言葉。もう，うんざりしてしまうかも…。
そんなに，一生懸命に覚えなくても，いいです。なんとなく，頭の中のすみっこの方に，覚えていてくれるだけで，十分。
うーん，実は，わたしも，きちんと覚えていません。スタッフの方に尋ねたり，自分で調べたり。

なので，安心してください，ね。

マイコンは，指示(しじ)された **命令の通りに動く** ように，作られています。

例えば。

「目玉焼を作ってね 」 と命令すれば，目玉焼を。**「チーズケーキを作って 」** と命令すれば，チーズケーキを作ります。（これは，例えばです！ マイコンは 目玉焼やチーズケーキを作ったりしません。ほんと残念です）。

しかも。

命令にそむいたりしません。

「目玉焼を 10 億コ，作ってね」と命令すれば，一生懸命(けんめい)に働いて目玉焼を 10 億コ作ります。「チーズケーキを $10^{10^{50}}$*コ，作って」と命令すれば，もう，しゃかりきに働いて，$10^{10^{50}}$コ作ります。

人間や動物だったら，だめでしょうね。

もしも，わたしが「目玉焼を 10 億コ，作って」と言われたら…。

そうですよね。たぶん，

「わたし，オムレツを作るの，上手です。だから，目玉焼 10 億コの代わりに，10 億倍おいしい！ オムレツ作ります」

とか言って，ごまかしてしまいます。（でも 10 億倍おいしいオムレツなんて，どんな味？）。

けれども，マイコンには 人間の言葉は理解(りかい)できません。「目玉焼を作ってね」と言っても，ぜんぜん分りません。

そこで。

マイコンが分る言葉を，用意します。それが

プログラム言語

だそうです。

● $10^{10^{50}}$
1兆を1兆コ用意しても，比べものにならない程，大きい数字です。

たくさん種類，あります

「プログラム言語」には，とてもたくさんの種類があります。例えば——。

	プログラム言語名
1	機械語*（マシン語）
2	アセンブリ言語*（アセンブラ）
3	FORTRAN*（フォートラン）
4	BASIC（ベーシック）
5	COBOL*（コボル）
6	PASCAL*（パスカル）
7	C*（シー）

これは，とても有名な「プログラム言語」だけ，だそうです。この中で，わたしが知ってるのは，**「機械語」**と**「BASIC」**。あとは，ほとんど名前を聞いたかな，程度です。

でも，まだまだ，たくさんの種類があるそうです。

機械語（machine language）

大型コンピュータやマイコンが，直接理解できる「言語」のことです。

機械語という呼び名の他に，マシン語，マシン・ランゲージ，マシン・コードという名前があります。

直接理解できる「言語」なので，とても高速です。『ファミリーコンピュータ』用のゲームカートリッジには，この機械語で作られたプログラムが記憶されているそうです。

●機械語

デジタル計算機は，いろいろな計算，処理や管理を一連の命令に従って行なう。この命令を人間が機械に伝えために使用されているのがプログラム言語である。デジタル計算機が直接理解できる言語は機械語（machine language）と呼ばれる。

機械語の命令は，単なる数字の系列であり，人間には理解しにくいとされている。

●アセンブリ言語

アセンブラという。

機械語の命令が，単なる数字の系列でしかなく，人間には理解しにくいのを補助するために開発された言語。機械の命令である数字を，人間が理解しやすいように意味のある言葉を略した記号を利用している。

●FORTRAN

FORTRAN（フォートラン）は，元々IBMの740というデジタル計算機システムのために開発された言語である。

しかし，IBMマシンが普及するにつれFORTRAN自身も普及。各計算機メーカーも，FORTRANを採用した。

●COBOL（コボル）

Common Business Oriented Language（事務用共通言語）を略したもの。

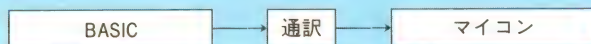
デジタル計算機の代表的なプログラム言語。事務計算などのデータ処理に適し，英語に似た形で書く。

BASIC（ベーシック）

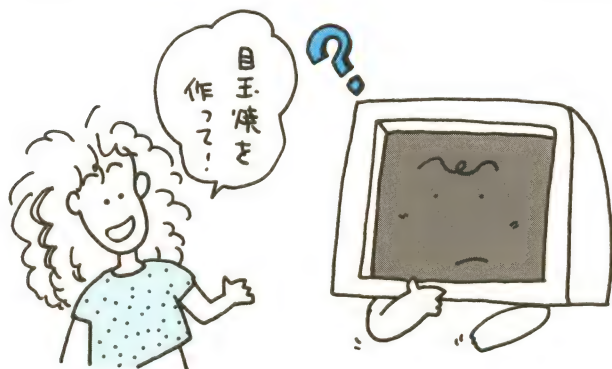
BASIC は，マイコンが直接理解できない「インタプリタ言語」*だそうです。
直接理解できないので，直接理解できる「マシン語」に翻訳しながら実行してゆきます。そのため，「マシン語」よりもスピードが遅くなってしまいます。
そのかわり，とても「プログラム」が作りやすくなっています。

『ファミリーコンピュータ』では　たくさんの「プログラム言語」の中から

	プログラム言語名
○	BASIC
△	機械語

の２つのプログラム言語が使えます。
ただし，「機械語」の方は，普通の人には使えないそうです。

● PASCAL（パスカル）
アルゴル系のデジタル計算機用プログラム言語。
マイコン用の言語としてはベーシックにつぐ普及。プログラミング教育用として作られたため，よく整理されており表現力に優れている。
その後開発された多くのプログラム言語に多大な影響を与えた。

● C（シー）
米ベル研究所で開発された言語。

● インタプリタ言語
言語を実行する時，擬似記号命令を機械語命令に通訳する言語のこと。

プログラムは行の集まりです。

この「2章」で，できたなら

2章で説明したいこと

1	ファミリーコンピュータの中身
2	ファミリーコンピュータの仕組み
3	BASIC
4	プログラム

というふうに，説明したいと思ってました。

そして，残るは

4	プログラム

だけです。

それでは

プログラム

について，説明しますね。

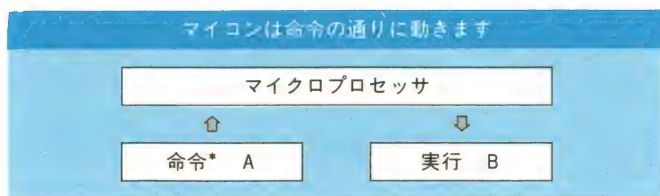
行が集まっています。

マイコンは，命令の通りに動くように，作られています。

マイコンは命令の通りに動きます

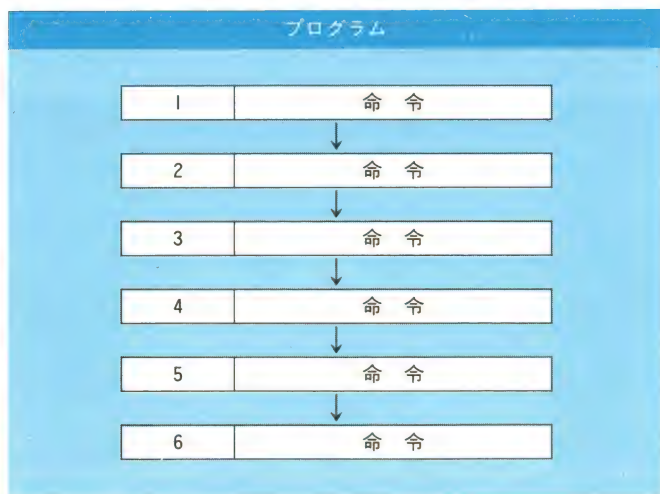
マイクロプロセッサ

⇧　⇩

命令*　A　　実行　B

このことは，前に説明しました。覚えていますか？

この

「命令」が集まったもの

が，プログラムです。

けれども「プログラム」は　命令が，**ごちゃごちゃに集まったもの**，ではないそうです。

「どんなふうに　並(なら)んでるの」

「ちゃんとした　理由はある？」

不思議(ふしぎ)，に思ってしまいますよね。

うーん，とにかく。本物のBASICのプログラムを，見てみましょう。

● 命令

instruction（インストラクション），command（コマンド）ともいう。

各段階で何の機能を行なうかということをデジタル計算機に知らせるプログラムの部分をなす。

命令は，デジタル計算機への指令を表わすいくつかの部分に分けることのできる一連の文字からなる。

● 行

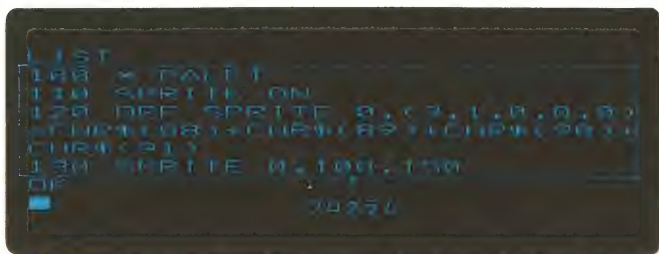
行は BASIC プログラムの基本的単位である。

BASIC の行の書式は，次の通り。

行番号　BASIC ステートメント[：BASIC ステートメント…]<キャリッジリターン>

プログラムは，1行に1つ以上の BASIC ステートメント（命令）を入れることができるが，それぞれのステートメントはコロン（：）で区切られている必要がある。BASIC の行は，常に行番号ではじまり，キャリッジリターンで終わる。最大255文字までを入れることが可能。

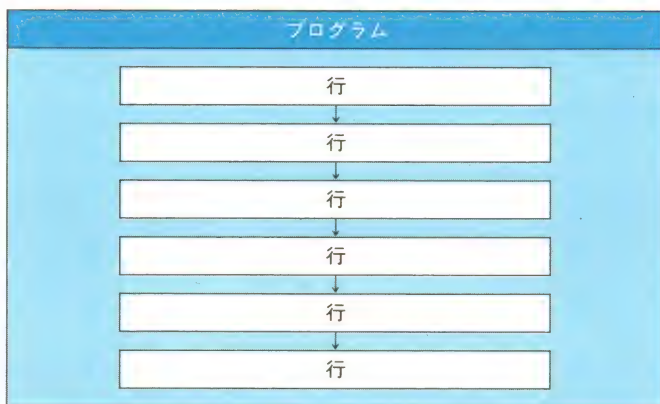
● 行番号

BASIC の行は，行番号ではじまる。行番号はメモリにプログラムの行を格納する順序を示し，また分岐や編集の目印としても使われる。

どうですか。

何となく，「命令」は**理由**があって　並(なら)んでいそうです。

「プログラム」は　まず

「行」*

という単位に，分けられます。

「行」は

行番号*と幾つかの「命令」

で，構成されているそうです。

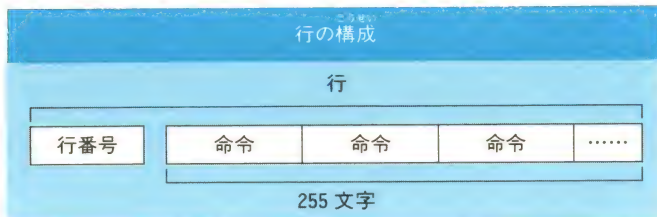
1つの行に，どれだけの命令を入れることができるのでしょうね。

でも，「幾つか」では，よく分りません。

こんなふうに，疑問(ぎもん)に思った人。『**ファミリーベーシック取扱説明書**(とりあつかい)』の **49** ページを，見てください。

49 ページには，次の図があります。

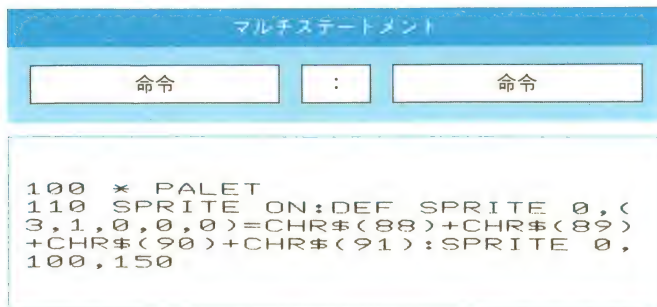
仕様	内　容
変数名の種類	英字で始まる先頭 2 文字を識別。 長さ 225 文字まで。
行番号の範囲	Ⓘ 265535
テキスト行の最大桁	255 桁

ですから，１行の文字数（桁数）が，　**255文字を超えなければ**，幾つでも入れることができます。

前のページの

```
100  * PALET
110  SPRITE  ON
120  DEF  SPRITE  0,(3,1,0,0,0)
=CHR$(88)+CHR$(89)+CHR$(90)+
CHR$(91)
130  SPRITE  0,100,150
```

というプログラムでは　**1「行」に1「命令」**　でした。

けれども　２つ以上の命令を入れる時は，かならず　命令と命令の間にコロン〈：〉を入れなくては，いけないそうです。

```
100  * PALET
110  SPRITE  ON:DEF  SPRITE  0,(
3,1,0,0,0)=CHR$(88)+CHR$(89)
+CHR$(90)+CHR$(91):SPRITE  0,
100,150
```

1つの「行」に，２つ以上の命令を入れることを

マルチステートメント

と，いいます。

順番に並びます

「行番号」に，どんな役目があるの，といいますと。

だいたい，

	内　容
①	行番号の小さい順に並ぶ役目
②	それぞれの行を区別する役目

という役目が，あります。

まず，①の **「行番号の小さい順に並ぶ役目」** について，説明したいと，思います。

「行」は，その

行番号の値の小さい順に並び

ます。

そこで，**「問題」** です。

もんだい 1	行の並び方

```
10 A=150
20 PRINT A
```

上のようなプログラムがあったとします。この時，次のような「行」を入力すると，どうなると思いますか。

```
15 A=A+A
```

「問題」と言っても，ぜんぜん，たいしたことありませんね。

みなさんは　すぐに「分ってしまった」って，思います。もうちょっと難しい問題の方が，良かったかも…。

一応，問題の答えは

こたえ 1	行の並び方

```
10 A=150
15 A=A+A
20 PRINT A
```

後から，新しい行を入力すると，プログラムの一番うしろに，くっついちゃうって感じしませんか。

朝礼の時，遅刻して来た子は，一番うしろに並びます。

でも「行」は，違います。

後から入力した「行」でも　行番号の順番に並びます。

だから，プログラムの途中に，新しい行を入れたい時，簡単にできます。

「行」は　順番に並ぶ

10	行
20	行
15	行
10	行
15	行
20	行

● LIST 命令

動　き

メモリ中のプログラムを画面に表示します。

文　法

LIST [m] [－[n]] m→表示させる最初の行番号／n→表示する最後の行番号

説　明

LIST は，メモリ中のプログラムを画面上に出力させるコマンド（命令）で，メモリ中のプログラムを調べたり，修正したりするときに使います。行番号を指定することによって，プログラムの任意の部分を自由に表示することができます。

LIST m……m 番のみを表示

LIST m，または LIST m－……m 番以降を表示

LIST n, n または LIST m－n……m 番から n 番まで表示。

LIST, n または LIST－n……n 番までを表示

LIST……全体を表示。

● メモリ（memory）

記憶装置ともいう。コンピュータの内部記憶装置を指すのに，普通使われる。ただし，厳密な意味では，CPU により直接指定されるロケーションを意味する。ディスク等の補助記憶装置は，ふくまない。しかしながら，補助記憶装置もディスク・メモリというように呼ばれることもある。この時，内部記憶装置は，メイン・メモリと呼び，補助記憶装置と区別するのが一般的である。

行番号は，目印です

①の「行番号の小さい順に並ぶ役目」は，説明してしまいました。今度は，②の

	内　容
②	それぞれの行を　区別する役目

です。

この②の説明は，ちょっとやりにくい雰囲気，です。

なので，「例え」でやってみます。

例えば――。

BASIC には，**LIST***，という命令があります。

この「LIST」命令は

LIST	プログラムを画面に表示させる。
LIST　[Cm] [－[n]] ・m 表示させる最初の行番号 ・n 表示する最後の行番号	

と書いて，メモリ*中の「プログラム」を画面上に表示させるための命令です。でも，もしも，「行番号」が無かったら…。

「プログラム」の，どこからどこまでを，画面に表示させるのかわかりません。

どうです？

何となく，分ってもらえたでしょうか。

「行番号」は

「行」の位置

を，指定する役目もあるんです。

よくわかる最高水準BASIC入門講座

この章では、『ファミリーベーシックV2.0』の説明をします。しかも、ちゃんとプログラム言語理論にそって、分りやすく説明します。
「ファミリーベーシック」は、ゲームを作るのに優れた、扱いやすいプログラム言語です。でも、扱いやすいので、反対にプログラミングを学べずに終わってしまうことが、よくあります。この章をよく読んで、正統的なプログラミングをマスターしましょう。なお、この章は担当は木村香奈枝です。

こんなこと、説明します。

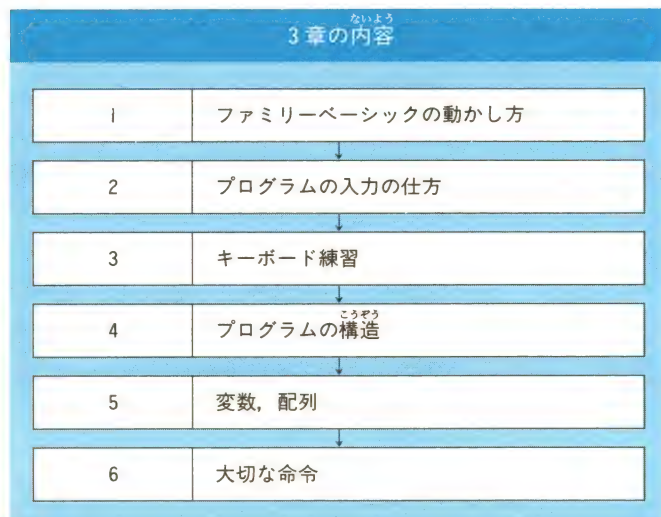
●ファミリーベーシック
ファミリーコンピュータ用BASICシステム。
任天堂㈱発売のファミリーコンピュータに接続し，NS-HuBASICを動かすことが出来る。NS-HuBASICは，㈱ハドソンソフトが中心となり開発。ゲームプログラムを制作するためのBASICである。

プログラム用ROM…8ビット×10K×2個／キャラクタゼネレータ用ROM…8ビット×8K／RAM…8ビット×2K／文字表示…28文字×24行（BASIC時）／表現色数…52色（白及，黒を含む）／スプライト…8スプライト，256×240ドット／サウンド…2個の方形波音，1個の三角波音，1個の効果音／キーボード…総数72キー

この「3章」では

ファミリーベーシック*

について，説明したいと思います。
だいたい，

3章の内容	
1	ファミリーベーシックの動かし方
2	プログラムの入力の仕方
3	キーボード練習
4	プログラムの構造
5	変数，配列
6	大切な命令

という内容になるといいな…。でも　なかなか，わたしの思った通りに，なってくれません。
みなさんは，この前の「2章」を読んで，『ファミリーコンピュータ』内部のこと，BASICのこと，プログラムのこと，なんとなく分りました？
『ファミリーコンピュータ』の中に，**CPU**や**ROM、RAM**なんていうものが入っているなんて…。
わたし，初めて知った時，「ファミリーコンピュータって　かしこい！」って，**感心しちゃいました。**
さあ，『ファミリーコンピュータ』のことが分ったら，今度は，『ファミリーベーシック』の番，です。

BASICの動かし方です。

それでは　初めに

『ファミリーベーシック』の動かし方

を，説明します。

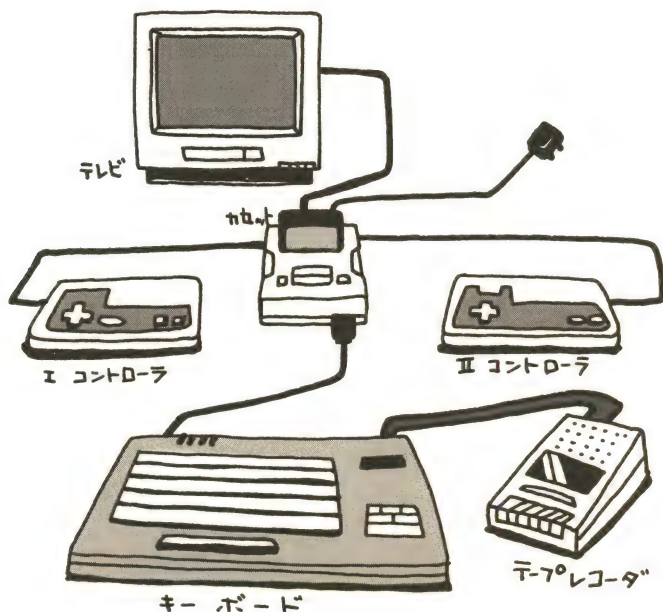
『ファミリーベーシック』の動かし方，**取扱説明書**（とりあつかい）*の２～７ページを読めば，とてもよく分ると思います。

でも。

「マニュアル，なくしちゃった」

「説明書を見るのは　面倒（めんどう）。」

という人がいるかも…。うーん，簡単（かんたん）に説明した方が，いいですね。

ファミリーコンピュータは，『キーボード*』と『カセット』を接続（せつぞく）すると　下の図のようになります。

●取扱説明書
ここでは,「ファミリーベーシック」の取扱説明書のことです。
ファミリーベーシックを買うと，かならず付いて来ます。これをなくすと，とっても困ってしまいます。
大切にしてあげてくださいね。

●キーボード（keyboad）
英数字，特殊記号をキーとして規則的に配置したもので，1つのキーを選択することによって，これに対応した信号を計算機などに送り出すもの。
0～9の数字からなるテンキーと英数字および記号からなるフルキーがある。

ファミリーベーシックの構成

ファミリーコンピュータ	+	キーボード	+	カートリッジ

ファミリーベーシック

『キーボード』と『カートリッジ』は，次の通りに，順番に接続してください。間違って接続すると，大切な『ファミリーコンピュータ』が壊れてしまいます。

特に，電源を入れたままで，接続しないように，とのことです。まず，**間違いなく壊れてしまう**そうです。

壊れちゃったら，たいへん。なので　かならず**電源を切ってから，**接続しましょうね。お願いします。

ファミリーベーシックを動かすには…

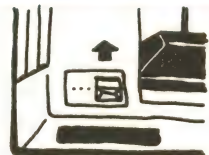
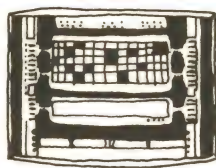
1　キーボードのコネクタを差し込む

電源をOFFにしたファミリーコンピュータのエキスパンドコネクターにキーボードを接続します。

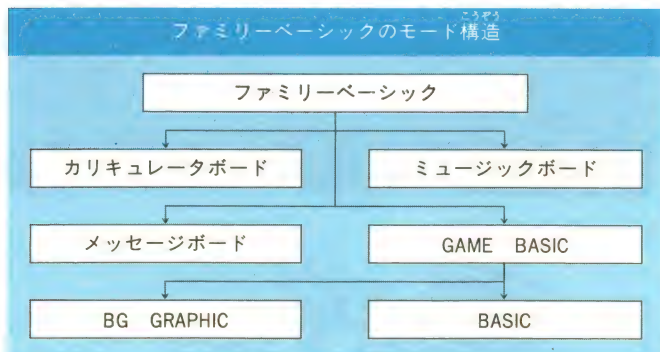
2　カセットを挿入する

カセットをファミリーコンピュータのカセットコネクタに挿入します。

3　電源をONにする

ファミリーコンピュータの電源を入れます。

4	スタート画面
スタート画面でコンピュータのカラーパネルが表示されます。	

BASICを選びます

これで，『ファミリーベーシック』が，動き出しました。
でも，まだ，やらなくちゃいけないこと，あります。『ファミリーベーシック』には，たくさんのモードがあります。もちろん，いろんな遊び方ができるように，です。

	ファミリーベーシックのモード
1	●カリキュレータボード画面*
2	●ミュージックボード画面*
3	●メッセージボード画面*
4	●GAME　BASICモード画面*
4—1	○BASIC*
4—2	○BG　GRAPHIC*

●カリキュレータボード
羽根ペンが動いて，計算式を表示する。ヨコ24文字×タテ8行を使用して計算を行う。

●ミュージックボード
音階の入力欄は上段，中段，下段に別れている。それぞれの段を利用し3オクターブの音を出力することが可能。

●メッセージボード
羽根ペンが動き，メッセージを表示する。

● GAME　BASIC モード
BASIC，BG　GRAPHIC の選択を行う。

● BASIC
BASIC 言語を使用して，プログラムを制作する。

● BG　GRAPHIC
ゲームの背景を描くためのグラフィック・エディタ。

●カーソル（Cursor）
CRT 画面上の文字位置を指定するための記号。
通常,〈■〉や〈_〉を利用，さらにはそれらを点滅させているものもある。
カーソルキーなどにより，移動させることができる。

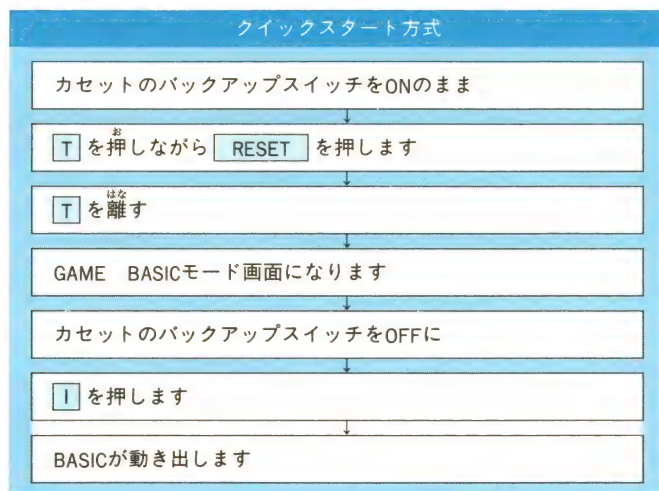
このモードの選択の仕方は，**取扱説明書の6～7ページ**に，くわしく載ってます。

それでは，たくさんのモードの中から

BASIC

を選びたい，と思います。

「BASIC」を選ぶ方法は，たくさんあります。

でも，わたしは，次の方法をおすすめします。

この方法は，一番簡単に「BASIC を選べる」そうです。この時，画面は次のようになっています。

入力は、難しくないです。

「BASIC」が，動きました。ホッと一息。

ところが。

BASICは「プログラム*」が無いと，ぜんぜん何もしません。何もしないのは，困っちゃいます。

そこで　今度は

プログラムの入力の仕方

について説明したいと思います。

プログラムは

プログラムの構造

行番号	行*
行番号	行
行番号	行
行番号	行
行番号	行
行番号	行

という構造になっています。

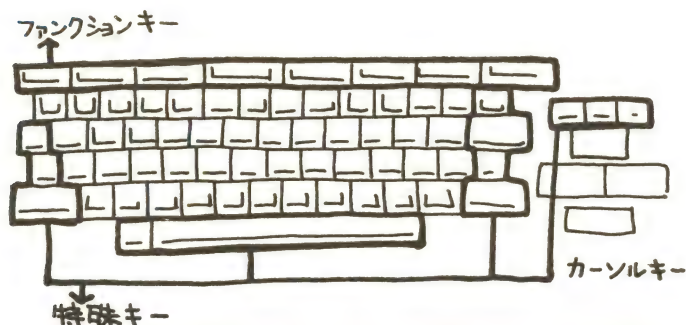
みなさん，覚えていてくれました？

「あ，忘れてた！」

なんて，人。もう，そんな人，**わたしホッペしちゃいます。**

キーボードの使い方，です

この「プログラム」を，実際に入力してみましょう。

キーボードの使い方は，**取扱説明書の4〜5ページ**を見てください。

文字キー	数字，英大文字，カタカナ，記号

英・数字	カーソル〈■〉
キーを そのまま	キーの左にかかれてある数字と英大文字や記号が入力できます。
SHIFT 押しながら	キーの上にかかれてある記号が入力できます。

カナ 押す　　カナ 押す

カナ	カーソル〈_〉
キーを そのまま	キーの下にかかれてあるカタカナ(大文字)や記号が入力できます。
SHIFT 押しながら	キーの右にかかれているカタカナ(小文字)や記号，はれつ音を入力。
GDPH 押しながら	カタカナの濁音とグラフィック記号が入力できます。

ファンクションキー

このファンクションキーを押す と，定義されている文字が入力されます。ですから，よく使う文字列をセットしておけば，プログラムやデータなどを入力する時間がずっと短くなります。各ファンクションキーは，最大 15 文字まで定義できますが，NS-HuBASIC を使用している時間，次のように定義されています。

カーソルキー

カーソルを移動する時に使います。

特殊キー

ESC	エスケープキー
カリキュレータ，ミュージック，メッセージ各ボードからセレクト画面に戻るとき等に使います。	

CTR	コントロールキー
GAME BASIC モードで指定のキーとの組み合わせていろいろなコントロールができます。	

SHIFT	シフトキー
文字キーと組み合わせて，カタカナの小文字やキーの上と右にかかれている記号が入力できます。	

GRPH	グラフィックキー
文字キーと組み合わせて，カタカナの濁音が入力できます。グラフィック記号（文字キーの最上列のキーのみ）が入力できます。	

	スペースキー
文字と文字の間に 1 文字文の空白（スペース）をつくります SPACE と書くこともある。	

STOP	ストップキー

BASIC プログラムの実行を中断（BREAK）するときに使います。
スタート画面，セレクト画面の表示を一時止める働きをします。
BG GRAPHIC の画面から GAME BASICモードに戻るときに使います。

カナ	カナキー

キーの下や右にかかれてあるカタカナを入力するときに使います。一度押すとカナ状態になり，もう一度押すと解除されます。

RETURN	リターンキー

入力した文字や記号がメモリされると同時にカーソルが次の行の先頭に戻ります。

CLR HOME	クリアホームキー

そのまま押すと画面の表示は消えずにカーソルはホーム位置（画面左上）に戻ります。
SFIFT を押しながら押すと画面の表示が消えて，カーソルがホーム位置に戻ります。

1行ずつ，入力します

「キーボード」には，72個ものキーがあります。

「こんなに,覚えられるかな」

と，不安になってしまうかも。でも，安心してくださいね。誰でも，みんな，そうなんです。慣れちゃえば，もう，何でもなくなってしまいます。

それでは，

プログラムの入力の仕方

を説明，します。

「プログラム」は

「行」単位で入力

します。ですから，プログラムを**1行ずつ**，キーボードから入力*していきます。

そして　一行入力したら，[RETURN] キー*を押してください。

RETURNキー*

これは　BASICに，**「1行入力しましたよ。覚えてくださいね」**と，知ってもらうため。[RETURN]キーを押さないと，　覚えてくれません。注意してくださいね。

こうして　すべての行を入力してしまえば　終りです。

プログラムの入力法

1行を入力します	RETURN
1行を入力します	RETURN
1行を入力します	RETURN
1行を入力します	RETURN
1行を入力します	RETURN

これだけの説明じゃあ，分りにくい，かも…。

そこで。

わたし，プログラム入力の実演(じつえん)をしてみます。なかなかのグッドアイデア，だと思いません？

説明するより，実際(じっさい)にお見せする方が，好き，です。「説明する」のって，難(むずか)しいですし。あまり居心地(いごこち)よくないです。

でも，実際にやってみせるのなら，**とっても楽。**ふだんの通り，ふつうにするだけだもん。

●入力

インプット（input）ともいう。

周辺機器からメモリへのデータやプログラム命令を転送するプロセスのこと。データそのものを指すのにも使われ，時には回路や装置に加えられる信号を指すのにも使われる。

●キャリッジ・リターン

キャリッジ・リターン（carriage return）は復帰，キャリッジ・リターン，CRともいう。

自動キャリッジをあらかじめ設定してある左端のマージンにまで戻して，次の文字を印字できるようにする操作のこと。

● SHIFT ＋
SHIFT キーを押しながら，
という意味です。

では　次の「プログラム」を入力してみますね。

```
100  SPRITE ON:DIM A$(1)
200  A$(0)="ケコサシ":A$(1)="コケシサ
"
210  FOR I=0 TO 99
220  FOR J=0 TO 3
230  DEF SPRITE J,(0,1,0,I MO
D 2,0)=A$(I MOD 2)
240  SPRITE J,100+J*16,100
250  NEXT:PLAY "T103CB"
300  FOR J=1 TO 500:NEXT
310  NEXT
```

うーん，けっこう沢山(たくさん)のキーを入力しますね。

自分でキーボードから入力した時，思いませんでした。でも，図にすると，**こんなに沢山になるので，びっくり……。**

この時の「画面」は，こんなふうになってます。

```
100 SPRITE ON:DIM A$(1)
200 A$(0)="ケコサラ":A$(1)="コケシサ
"
210 FOR I=0 TO 99
220 FOR J=0 TO 3
230 DEF SPRITE J,(0,1,0,1 MO
D 2,0)=A$(I MOD 2)
240 SPRITE J,100+J*16,100
250 NEXT:PLAY "T103CB"
300 FOR J=1 TO 500:NEXT
310 NEXT
```

プログラムを入力する時，「気をつけてほしいな」と思うことがあます。

プログムは，「行」ごとに入力します。

でも　この「行」は**画面の行のことではなくて，**

プログラムの「行」

のことです。

例えば，次のようなプログラムがあります。

```
100 SPRITE ON:DEF SPRITE 0,(
0,1,0,0,0)="ンラァィ":SPRITE 0,1
00,100
```

●左のプログラム
この左にある BASIC のプログラムは，画面中央に「青いレーシングカー」を表示させるものです。
気が向いたら，試してみてくださいね。

このプログラム，**画面の行数**は「3行」。でもプログラムの**行数**では「1行」です。

画面の行とプログラムの行，ごっちゃになりやすいので，気をつけてくださいね。

● RUN（ラン）

働　き

プログラムを実行させます。

文　法

RUN　n

n→実行を開始したい行の行番号

説　明

RUN は，プログラムを実行させるためのコマンドです。RUN を入力すると，プログラムは最初から順に実行されますが，このとき変数はすべてクリアされます。

また，プログラムを途中から実行させたい場合は，実行開始行番号 n を付けます。

STOP を押してプログラムをよむためと，再度実行するときに RUN や RUN n を使用すると，それまでの変数に入っていた内容をすべてクリアしてしまうので，変数クリアせずに実行したい場合，GOTO や CONT を使用してください。

もったいない，です

さっきのプログラム。

そのままにしておくの，ちょっともったいないです。せっかくだから，動かしてみましょう。

BASIC のプログラムを動かすには

「RUN*」命令

を使います。

詳(くわ)しくは，後で説明しますから，**「BASICプログラムを動かすための命令なんだな」**と覚えておくくらいで，十分です。

次の図の通りに，入力すれば動きます。

すると。

画面の中央で，4匹(ひき)のペンギンさんが「ペコペコ」踊(おど)り出します。

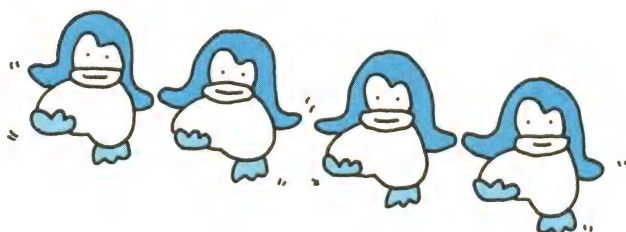
短いプログラムのわりに，とても面白くできました。4匹(ひき)のペンギンさんがラインダンスをするなんて…。

そうそう，忘(わす)れていました。

プログラムのリストを画面に表示(ひょうじ)させる

「LIST*」命令

LIST	プログラムリストを表示します
L I S T RETURN	

というのもあります。一度，使ってみるのもいいと思います。

● LIST（リスト）

動　き

メモリ中のプログラムを画面に表示します。

文　法

LIST [[m] [−[n]] m→表示させる最初の行番号／n→表示する最後の行番号

説　明

LIST は，メモリにのっているプログラムを画面上に出力させるコマンド（命令）で，メモリ中のプログラムを調べたり，修正したりするときに使います。行番号を指定することによって，プログラムの任意の部分を自由に表示することができます。

すごい，タッチメソッド

みなさんは，プログラムを入力する時，どうしています？　どんなふうに，入力していますか。

たぶん。

わたしの予想，です。「人差し指」かなにかで　ポチポチと入力しているんじゃあないでしょうか。

これ，なかなか大変です。1つの文字を入力するのに　おお騒ぎ。キーボードの「アルファベット」は　**ほとんどバラバラに配置されていますから**，一生懸命探します。

1つの文字を探すのに，20秒くらいかかったり…。

これでは，イヤになってしまうのも，無理ないです。

でも。

キーボードを見ないで，ものすごい速さで入力する人がいます。みなさんも，何かで見たり聞いたりしたことある，と思います。

あれは，

タッチメソッド

という，入力の仕方だそうです。

あんなに速く入力できたら，いいですね。

そこで，この「タッチメソッド」を説明したいと思います。

練習を始める前に

「タッチメソッド」は**タイピストの方のための入力方法です。**本格的にマスターするには　半年くらいかかってしまいます。

けれども　マイコンでは，本格的にマスターしなくても，大丈夫。

毎日20分，7日続ければ，十分です。

あとは，プログラムを入力すればするほど上達します。

練習をする前に，次の「練習を始める前に」を読みましょう。

練習を始める前に

1	先を急がないでください。
初めの2週間ほどの間は，タイピング技術の基礎を作るとても大切な練習期間です。先を急ぎ過ぎると，練習が不十分になったり，悪い癖がついたりします。	

2	キーボードを見ないように
視線はいつもテキストを見て，絶対にキーボードを見ないでください。キーの位置が分らない時は，キーボードの表を壁などに貼り，それを見るようにします。	

3	正確に，急がないように
正確に打つようにして，誤りを避けてください。初めの頃は，スピードよりも正確さを大切にします。 スピードを気にすると，キーボードを見たくなりますが，どんな場合でも見ないようにしてください。	

そして，次の「キーボード」の図を見てください…ね。

指の分担とホームポジション

この「指の分担とホームポジション」という図は，どの指がどのキーを押したらいいのかを，説明しています。

下から2番目のキーの中に，8つだけ「色」が違います。ここは**ホームポジション**といって，いつも指を乗せておきます。

9日でマスターできます

●注意（第1日目）
○練習中は，キーボードを見ないようにしてください。
○打ちにくい所は，自信のつくまで練習します。
○スペースを作る時は，左指で打ち終れば右手親指で，右手指で終ったら左手親指でスペースバーを打ちます。この方法は，シフトキーを使用する時に，役立ちます。

EXERCISE 1	1日目	人差指の練習

・問題をよく見て，ゆっくり打ちます。
・リズムを持ってゆっくり打つのが大切です。
・人差指以外は，なるべくホームポジションから離れないようにします。

```
JJJJ  JJJJ  JJJJ  JJJJ  JJJJ
FFFF  FFFF  FFFF  FFFF  FFFF

JFJF  JFJF  JFJF  JFJF  JFJF
FJFJ  FJFJ  FJFJ  FJFJ  FJFJ

HHHH  HHHH  HHHH  HHHH  HHHH
GGGG  GGGG  GGGG  GGGG  GGGG

HGHG  HGHG  HGHG  HGHG  HGHG
GHGH  GHGH  GHGH  GHGH  GHGH

UUUU  UUUU  UUUU  UUUU  UUUU
YYYY  YYYY  YYYY  YYYY  YYYY

JUJU  JUJU  JUJU  JUJU  JUJU
JYJY  JYJY  JYJY  JYJY  JYJY
```

●注意（第2日目）
○ホームポジションから離れた指は，そのつど元に戻します。たとえ打つ同じ字（G）であっても，打つたびに元に戻します。今は，ホームポジションを基本にして他の文字の位置をおぼえる練習です。
○手首はキーボードに対して，なるべく水平にするようにします。

EXERCISE 2	2日目	続・人差指の練習

・昨日に続いて，人差指の練習です。
・もう一度，昨日練習したキーを打ってみましょう。

```
RRRR  RRRR  RRRR  RRRR  RRRR
TTTT  TTTT  TTTT  TTTT  TTTT

FRFR  FRFR  FRFR  FRFR  FRFR
FTFT  FTFT  FTFT  FTFT  FTFT

MMMM  MMMM  MMMM  MMMM  MMMM
NNNN  NNNN  NNNN  NNNN  NNNN

JMJM  JMJM  JMJM  JMJM  JMJM
JNJN  JNJN  JNJN  JNJN  JNJN

VVVV  VVVV  VVVV  VVVV  VVVV
BBBB  BBBB  BBBB  BBBB  BBBB

FVFV  FVFV  FVFV  FVFV  FVFV
FBFB  FBFB  FBFB  FBFB  FBFB
JMJV  JMJV  JMJV  JMJV  JMJV
JNJB  JNJB  JNJB  JNJB  JNJB

FURV  FURV  FURV  FURV  FURV
HURT  HURT  HURT  HURT  HURT
GRUM  GRUM  GRUM  GRUM  GRUM
FRUM  FRUM  FRUM  FRUM  FRUM
```

EXERCISE 3 第3日目 中指の練習

• 今日から中指の練習です。順次，日頃(ひごろ)ほとんど使っていない指の練習になりますから，あわてずゆっくり，そしてリズムを忘れずに練習します。
• 人差指は軽くホームポジションに残っているように注意します。
• 親指は，スペースバーのところに置きます。
• 新しい字の練習には，ゆっくり練習します。

```
KKKK  KKKK  KKKK  KKKK  KKKK
DDDD  DDDD  DDDD  DDDD  DDDD

KDKD  KDKD  KDKD  KDKD  KDKD
DKDK  DKDK  DKDK  DKDK  DKDK

IIII  IIII  IIII  IIII  IIII
EEEE  EEEE  EEEE  EEEE  EEEE

IEIE  IEIE  IEIE  IEIE  IEIE
EIEI  EIEI  EIEI  EIEI  EIEI

,,,,  ,,,,  ,,,,  ,,,,  ,,,,
CCCC  CCCC  CCCC  CCCC  CCCC

,C,C  ,C,C  ,C,C  ,C,C  ,C,C
C,C,  C,C,  C,C,  C,C,  C,C,
```

●注意（3日目）

○中指は強い指ですから，練習もいくぶん楽にできますが，この指はよく使われるeとiを受け持っています。

人によっては，左右の指の錯覚を起こし間違えたりすることもあるようです。eとiとを区別して，よく覚えておきます。

EXERCISE 4 第4日目 人差指と中指の練習

• 次は人差指に中指の加わった練習です。

```
JKIK  JKIK  JKIK  JKIK  JKIK
FEDE  FEDE  FEDE  FEDE  FEDE
JKIK  JKIK  JKIK  JKIK  JKIK

K,K,  K,K,  K,K,  K,K,  K,K,
DCDC  DCDC  DCDC  DCDC  DCDC

THE  THE  THE  THE  THE  THE
BIG  BIG  BIG  BIG  BIG  BIG

CUT  CUT  CUT  CUT  CUT  CUT
TIE  TIE  TIE  TIE  TIE  TIE

BUY  BUY  BUY  BUY  BUY  BUY
YET  YET  YET  YET  YET  YET

THEY  THEY  THEY  THEY  TYEY
FINE  FINE  FINE  FINE  FINE

REND  REND  REND  REND  REND
HIBR  HIBR  HIBR  HIBR  HIBR

VERY  VERY  VERY  VERY  VERY
MICE  MICE  MICE  MICE  MICE
```

●注意（第5日目）

○体が曲がったり，ひじが張ったりしないように気を付けます。

○ S，W，X は打ちにくく間違いやすい文字です。チェックしながら自信をつけましょう。

EXERCISE 5　第5日目　薬指の練習

・薬指はいちばん弱い指ですから一生懸命(けんめい)練習してください。

・薬指の場合，他の指がどうしても浮(う)きがちですが，なるべくホームポジションに置くようにします。難(むずか)しければ，小指をホームポジションに乗せて，人差指と中指を少し浮かしてもいいです。

```
LLLL  LLLL  LLLL  LLLL  LLLL
SSSS  SSSS  SSSS  SSSS  SSSS

LSLS  LSLS  LSLS  LSLS  LSLS
SLSL  SLSL  SLSL  SLSL  SLSL

OOOO  OOOO  OOOO  OOOO  OOOO
WWWW  WWWW  WWWW  WWWW  WWWW

OWOW  OWOW  OWOW  OWOW  OWOW
WOWO  WOWO  WOWO  WOWO  WOWO

....  ....  ....  ....  ....
XXXX  XXXX  XXXX  XXXX  XXXX

.X.X  .X.X  .X.X  .X.X  .X.X
X.X.  X.X.  X.X.  X.X.  X.X.
```

EXERCISE 6　第6日目　人差指・中指・薬指

・今まで練習に加え，薬指の練習をします。

・誤(あやま)りは原則として避(さ)けたいですが，あまり神経質(しんけいしつ)になる必要はありません。

・いつもキーの中央を打つように。

```
JKLJ  JKLJ  JKLJ  JKLJ  JKLJ
FDSF  FDSF  FDSF  FDSF  FDSF

JLOL  JLOL  JLOL  JLOL  JLOL
FSWS  FSWS  FSWS  FSWS  FSWS

JL.L  JL.L  JL.L  JL.L  JL.L
FSXS  FSXS  FSXS  FSXS  FSXF

ONLY  ONLY  ONLY  ONLY  ONLY
MILK  MILK  MILK  MILK  MILK
BEST  BEST  BEST  BEST  BEST

USED  USED  USED  USED  USED
KNOM  KNOW  KNOW  KNOW  KNOW
VIEW  VIEW  VIEW  VIEW  VIEW

NEXT  NEXT  NEXT  NEXT  NEXT
COXY  COXY  COXY  COXY  COXY

WORLD  WORLD  WORLD  WORLD
CHILD  CHILD  CHILD  CHILD
CLOSE  CLOSE  CLOSE  CLOSE
WORSE  WORSE  WORSE  WORSE
```

EXERCISE 7 第7日目 小指の練習

•小指は，ホームポジションの基本指を使うことになりますから最も重要です。小指の練習をして，慣れるにつれて他の指もホームポジションから浮かせてもよいのです。その時には，小指がホームポジションの基本指となります。

```
;;;;  ;;;;  ;;;;  ;;;;  ;;;;
AAAA  AAAA  AAAA  AAAA  AAAA

;A;A  ;A;A  ;A;A  ;A;A  ;A;A
A;A;  A;A;  A;A;  A;A;  A;A;

PPPP  PPPP  PPPP  PPPP  PPPP
QQQQ  QQQQ  QQQQ  QQQQ  QQQQ

PQPQ  PQPQ  PQPQ  PQPQ  PQPQ
QPQP  QPQP  QPQP  QPQP  QPQP

////  ////  ////  ////  ////
ZZZZ  ZZZZ  ZZZZ  ZZZZ  ZZZZ

/Z/Z  /Z/Z  /Z/Z  /Z/Z  /Z/Z
Z/Z/  Z/Z/  Z/Z/  Z/Z/  Z/Z/
```

●注意（第7日目）

○小指は力が弱く，特に左手小指は一番使い慣れない指です。他の指の数倍，練習します。

○Ａは　とても多く使用される文字ですので，自信がつくまで練習してください。

○小指は使い慣れていませんので，他の指で代用しやすいものです。根気よく忠実に打ってください。

EXERCISE 8 第8日目 すべての指の練習

```
JKL;  JKL;  JKL;  JKL;  JKL;
FDSA  FDSA  FDSA  FDSA  FDSA

J;P;  J;P;  J;P;  J;P;  J;P;
FAQA  FAQA  FAQA  FAQA  FAQA

J;/;  J;/;  J;/;  J;/;  J;/;
FAZA  FAZA  FAZA  FAZA  FAZA

PUMP  PUMP  PUMP  PUMP  PUMP
QUAY  QUAY  QUAY  QUAY  QUAY
SIZE  SIZE  SIZE  SIZE  SIZE

QUIZ  QUIZ  QUIZ  QUIZ  QUIZ
PLACE  PLACE  PLACE  PLACE
QUITE  QUITE  QUITE  QUITE

DOZEN  DOZEN  DOZEN  DOZEN
SPENT  SPENT  SPENT  SPENT
REPLY  REPLY  REPLY  REPLY

SPEAK  SPEAD  SPEAK  SPEAK
PIECE  PIECE  PIECE  PIECE
QUEEN  QUEEN  QUEEN  QUEEN
```

●注意（第8日目）

○基本的な心得に従って練習できていますか。悪い癖がないですか。もしも癖があったら，なおしましょう。

○ホームポジションは，覚えましたか。打つ前に，ホームポジションを想像してみてください。

○キーボードを見ないで，両手を同時にホームポジションにのせてみます。きちんと乗っているか確かめてください。

EXERCISE 9　第9日目　数字の練習

・数字は困難(こんなん)な練習です。初歩のうちに十分練習しておきます。
・反射(はんしゃ)的に動くようになるまで練習し，ちょくちょく練習しましょう。

```
JU8  JU8  JU8  JU8  JU8  JU8
HY7  HY7  HY7  HY7  HY7  HY7

FR5  FR5  FR5  FR5  FR5  FR5
GT6  GT6  GT6  GT6  GT6  GT6

KI9  KI9  KI9  KI9  KI9  KI9
DE4  DE4  DE4  DE4  DE4  DE4

DE3  DE3  DE3  DE3  DE3  DE3
LO0  LO0  LO0  LO0  LO0  LO0

SW2  SW2  SW2  SW2  SW2  SW2
AQ1  AQ1  AQ1  AQ1  AQ1  AQ1

24680  24680  24680  24680
13569  13579  13579  13579

3456  3456  3456  3456  3456
8675  5768  8675  5768  8675
```

CTRも使ってみましょう

このテキストを練習すれば，上手にキーボードを打てるようになる，と思います。

でも。

もっと上手になりたい人は

コントロールキー

を使うといいそうです。

コントロールキーを使うと

	コントロールキーを使う理由
1	より速くタイプできるようになります。
2	他のマイコンを使う時，楽です。

という利点が，あります。

コントロールキーは，それ１つだけでは，何もできません。他のキーと組み合せて使います。

CTR	コントロールキー
他のキーと組み合わせて，いろいろな働きをします。	

CTR ＋	処理内容(ないよう)
A	INS モードの ON/OFF スイッチ
C	BREAK。ただし，プログラム実行中はききません。
D	初期設定(せってい) CGEN2, SPRITE OFF, CTR ＋ A の解除(かいじょ)。カラーパレットをスプライト，バックグラウンドと共に，バックグラウンド用のパレットコードにする。
E	カーソル以降(いこう) 1 行分の消去。
G	BEEP 音を出します。
H	DEL と同じ機能(きのう)です。
J	ラインフィード（行送り）をします。▼ と同じです。

●ホームポジション
画面左上の位置のことをいう。

●ホームポジション
キーボードから入力する場合に，指を常に置いておく位置のこと。
左手は，小指→A，薬指→S，中指→D，人差指→F，に置き，右手は，小指→；，薬指→L，中指→K，人差指→J，に置く。

キー	機能
K	カーソルをホームポジションに戻します。
L	画面のクリア。
M	1行入力して改行。RETURN と同じ。
R	INS と同じです。
V	カナモードにする。
W	英・数字状態にする。
Z	カーソル以降画面下までクリアします。

「もっと速くキーボードから入力できるようになりたい」と思う人は，このコントロールキーの組み合わせの中から CTR＋H，CTR＋M，CTR＋R，の3つを使うようにすると，いいそうです。

DEL や INS キーを押す時，手が「ホームポジション*」から離れてしまいます。「ホームポジション」から手が離れると，とても時間がかかってしまいます。

そこで，DEL や INS キーの代りに，CTR＋H や CID＋M を使います。「コントロールキー」は，ホームポジションに手を置いたまま，入力できるから，パッパッと入力できるんです。

遊びながら、入力上手になれます。

「キーボード」の練習をすることは　とても大切なことって思います。

でも，ちょっぴりあきてしまいます…ね。

「練習」は毎日，同じことを繰り返します。同じことを何回も繰り返さなくては　練習にならないので　しかたないです。

毎日，まいにち，

```
JKIK  JKIK  JKIK  JKIK  JKIK
FEDE  FEDE  FEDE  FEDE  FEDE
JK■
```

なんてやっていたら，すぐに眠(ねむ)くなってしまいます。

そこで，みなさんの「眠け覚まし」に

遊んで練習するプログラム

を，作りました。わたし，**一生懸命，**作りました。

遊んでくれなくても　しかたありません。

でも，一人でも遊んでくれたなら…。とってもうれしいです。

このプログラムは，次のようにして動かします。

このプログラムの動かし方	
1	ファミリーベーシックを動かします。
2	BG GRAPIC を入力します。
3	BASIC プログラムを入力します。
4	実行します。

●コンピュータ起動方法

①キーボードのコネクターを差し込む…電源(POWER)を OFF にしたファミリーコンピュータのエキスパンドコネクタ（機能拡張コネクタ）にキーボードのコネクタを接続します。

②カセットを挿入します…カセットをファミリーコンピュータのカセットコネクタに挿入します。

③電源を ON にする…ファミリーコンピュータの電源（POWER）を入れます。

④スタート…スタート画面でコンピュータのカラーパネルが表われます。

・BG GRAPHIC 選択方法（クイックスタート）

①カセットのバックアップスイッチを ON のままで

②キーボードの[T]キーを押しながらファミリーコンピュータの[RESET]スイッチを押し[T]キーを離します。

③この結果，GAME BASIC モード画面が表示されますので，カセットのバックアップスイッチを OFF にしてください。

それでは『ファミリーベーシック』を動かして

BASICプログラムを入力

しましょう。

この本の終わり(186ページ)にある**「BG GRAPHIC 背景表」**を見て，その通りに入力します。

「BG　GRAPHIC」の操作方法は　次のようになってます。

BG　GRAPHIC 操作手順

BG　GRAPHIC の最初のモードは SELECT です。

↓

[CLR] または [SHIFT] ＋ [CLR] を押して，8 個のグループを選びます。

↓

[INS]，[DEL] を押して，キャラクタを選びます。

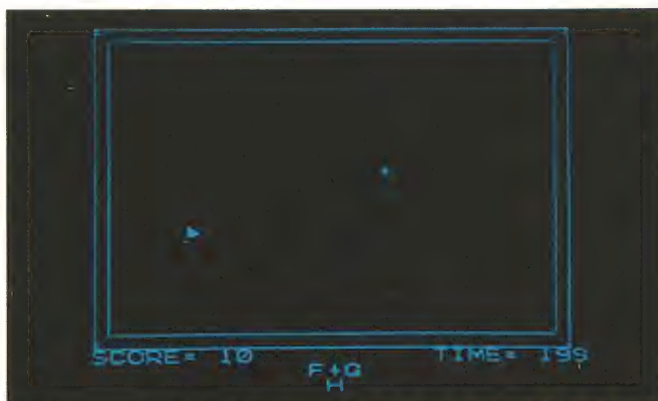
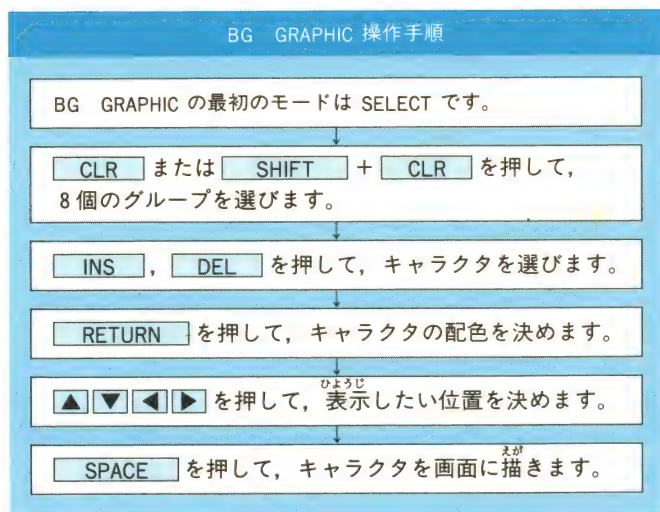
↓

[RETURN] を押して，キャラクタの配色を決めます。

↓

[▲][▼][◀][▶] を押して，表示したい位置を決めます。

↓

[SPACE] を押して，キャラクタを画面に描きます。

「BG　GRAPHIC」で背景を描き終ったら　次は

BASICプログラムを入力

します。

下の図の「プログラム・リスト*」を　入力してくださいね。

```
10  DATA JFGH,ULJY,TRMN,JNBV
15  DATA FURV,HURT,FRUM,GRUM
20  DATA JMFB,FBEV,KDIE,DECI
25  DATA LSOW,LSXW,JLKD,KDSF
30  DATA JLOX,ONLY,MILK,BEST
35  DATA NEXT,COXY,KNOW,VIEW
40  DATA ;APQ,/ZA;,JLD;,FDSA
45  DATA PUMP,SIZE,FAQE,QUIZ
50  DATA 1543,6785,1390,9823
100  '* ガメンヲセッテイシマス *
110  INPUT"LEVEL(1-50)";L
120  CLS:CGEN 2:CGSET 1,0
130  VIEW:SPRITE ON
200  '* ガメンヲツクリマス *
210  LOCATE 0,21:PRINT"SCORE=
            TIME="
230  H=0:S=0:T=250
240  X=13:Y=18:A=0:B=-1
250  GOSUB 500
300  '  * プログラムヲツクリマス *
310  FOR T=250 TO 0 STEP -1
320  IF T MOD 25=0 THEN GOSUB
 600
330  I$=INKEY$
340  IF I$=A$ THEN B=-1:A=0
350  IF I$=B$ THEN A=-1:B=0
360  IF I$=C$ THEN A= 1:B=0
370  IF I$=D$ THEN B= 1:A=0
380  LOCATE X,Y:PRINT" "
390  X=X+A:Y=Y+B
400  IF SCR$(X,Y)<>" " THEN G
OSUB 800
410  LOCATE X,Y:PRINT CHR$(20
7):LOCATE 24,21:PRINT T
420  FOR W=0 TO L*10:NEXT
430  NEXT:GOTO 800
500  '* ハタヲタテマス *
510  T=T+30:S=S+1
520  M=RND(22)+3
530  N=RND(15)+3
540  LOCATE M,N:PRINT CHR$(19
9)
550  LOCATE 6,21:PRINT S
560  PLAY"T6EG"
570  RETURN
600  '* ホウコウヲキメマス *
610  READ J$:PLAY"T1E"
620  A$=MID$(J$,1,1)
630  B$=MID$(J$,2,1)
640  C$=MID$(J$,3,1)
650  D$=MID$(J$,4,1)
660  LOCATE 13,21:PRINT A$
670  LOCATE 12,22:PRINT B$+"+
"+C$
680  LOCATE 13,23:PRINT D$;
690  RETURN
800  '* ナニカニブツカッタ *
810  J$=SCR$(X,Y)
820  IF J$=CHR$(199) GOTO 500

830  LOCATE 5,10,PRINT"*** GA
ME OVER ***"
840  PLAY"CDGAC":PAUSE:RUN
900  '* セイサク=キムラカナエ *
```

● BASIC 起動方法

「BG・GRAPHIC」から，BASIC を起動するには，次のような手順で行います。

① ESC キーを押す
② STOP キーを押す
③その結果，GAME　BASIC モード画面になる
④ I を選択する。

●プログラム・リスト

プログラム・リスト（program list）は，リストともいう。プログラムを，プリンタなどにより印字したもののことである。

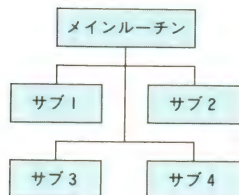
●プログラムの改良

いちおうゲームの難しさを設定できるようになってますので難しさを変えたい人は変数Tの値を増やしてください。

230行，310行にあります。ただし，25の倍数で増やしてください。

● データ（data）
何らかの条件値，または状態を示す数字。アルファベット文字，または記号からなる演算数，または因子からなるのが一般的である。例えば，デジタル計算機のプログラムによって演算される数値はデータであるが，プログラム自体はデータと呼ばない。
デジタル計算機による処理が可能なように変換された情報のこと。

● 変数
BASIC のプログラム中で使われる値（数値だけとは限らない）を格納するためのエリアに，英数字からなる名前を対応させたものである。

● メイン・ルーチン
メイン・ルーチン（main routine）は，主ルーチン，主プログラム，メイン・プログラムともいう。
プログラムの中心的な部分であり，処理されるデータの性質やプログラムの演算中に起こった条件に従ってサブルーチンへ制御を移す。

プログラムの説明

行番号	内容（ないよう）
10〜50	コンパスのデータ*
コンパス（？）を表示（ひょうじ）するためのデータです。	
100〜130	画面設定（せってい）
画面の設定や，いろんなことをしています。 110 行では，レベルの入力をします。	
200〜250	画面作成
本当は，変数*の設定を主にしています。 250 行では　ハタを表示します。	
300〜430	メインルーチン*
一番大切な「ボール」を動かしています。 320 行は，コンパスの設定をしていたり，330〜410 行では「ボール」を 1 つ動かしています。 特（とく）に，400 行はボールがぶつかったかを調べています。	
500〜570	ハタを表示（サブルーチン*）
ハタを画面に表示します。250 行と 82 行から呼（よ）ばれています。 この時，スコアやタイムの計算も行っています。	
600〜690	コンパスを表示（サブルーチン）
10 行〜50 行のデータを読み出し，それをコンパスとして画面に表示します。320 行から呼ばれています。	
800〜840	ゲームオーバー
ぶつかったものがハタであれば　500 行へ行きます。 そうでない時は，ゲームオーバーです。何かキーが押（お）されるまで待っています。	
900	注　釈（ちゅうしゃく）
せっかく一生懸命（けんめい）に作ったのですから，何か書こうと思って付けました。	

実行します

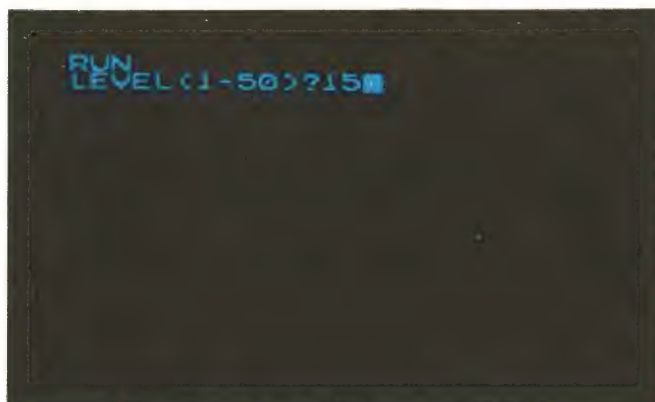
入力できましたか？

では，

実　行

してみたいと思います。

あ…その前に，「どんなゲームなのか」を説明しなくてはいけませんでした。もう，**わたし，忘れんぼさんなので困り ます。**

この「プログラム」を実行すると，画面は，次のようになります。

名　前	内　容
ボール	みなさんが動かすキャラクタです。
ハタ	消すと得点になります。
コンパス(?)	ボールを動かす時に，この指示されたキーを使います。
壁	ボールがここにぶつからないようにします。
得点	今までのスコア
残り時間	残り時間が無くなるとゲームオーバーです。

●サブルーチン(subroutine)
プログラム全体の機能のうち論理的にまとまった一部分も遂行し，それを構成する特定の命令群が必要なときには，いつでも利用できるようになっているプログラムの一部。

●注釈
プログラム内容を明確にするために書かれるコメントのこと。これは計算機に対しては，何の効果もない。

ゲームの内容は

ボールを動かしてハタを取る

というものです。ハタを取ると「得点」になり，残り時間も少し増えます。消されたハタは，画面のどこかに，現れます。

「な〜んだ簡単だ」

なんて，言わないでくださいね…。

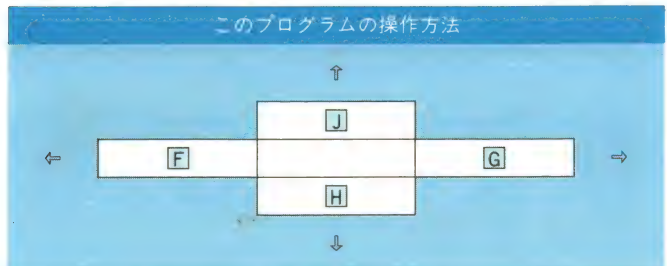
操作方法が，とっても難しいんです。

画面の下の方に

という，何だか「不思議」なもの，あります。

この「何だか不思議なもの」は，ボールを操作するキーの説明をしています。

ボールは，コントローラーで操作できません。その代りに，キーボードで操作します。

上図の場合，ボールを上に動かす→J，ボールを下に動かす→H，がボールを左に動かす→F，ボールをＥに動かす→G，を使って操作します。

しかも，この「コンパス」は，次々に変っていってしまいます。

ちょっと気を抜くと，**ゲームオーバー。**

けっこう難しいです。

ダイレクトとプログラム

今度は

プログラムの構造

について，説明したいと思います。

「プログラムの構造」なんて，とても難しいことを説明しなくちゃいけないみたいです。

でも，わたし，難しいことなんて，説明できないです。

「プログラム」を遠～くの方から見ると，こんなふうに見えます。こんなふうに，できてます。

それを，説明するだけ。それで，精一杯…。

プログラムの構造には

	プログラム構造の種類
1	ジャンプ（GOTO）*
2	分岐（IF～THEN…）*
3	繰り返し（FOR～NEXT）*
4	サブルーチン（GOSUB, RETURN）

の4種類，あるそうです。

● GOTO

働き　指定した行番号へ無条件にジャンプします。

文法　GOTO ｛行番号｝

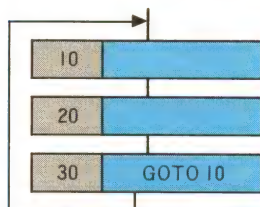
行番号→ジャンプ先の行番号

説明　GOTO 命令は，指定した行番号に無条件にジャンプするステートメント（命令）で，プログラムの流れを変えるときに使用します。

ダイレクト命令では，変数の値をクリアせずに，特定行番号からプログラムを実行することができます。

実行方法は，2種類あります

BASIC の動き方は，二つあります。

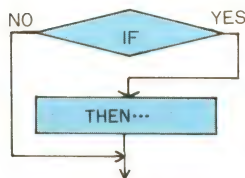
BASIC の動作の種類	
1	ダイレクトモード
行番号をつけずに BASIC の文法に添った文を入力した場合，その文は，キャリッジリターン入力後，すぐに実行されます。	

● IF～THEN…

働き　論理式による分岐を行います。

文法　IF　式　THEN｛行番号または命令｝

式→論理式／行番号→ジャンプ先の行番号／命令→任意の命令

説明　IF 文は，条件分岐を行うステートメントで，THEN と対にして使います。IF と THEN の間に論理式を書き，論理式が成立した場

合 THEN 以降を実行し，成立しなければ，次の行を実行します。

● FOR～TO―STEP…NEXT

働き FOR―NEXT ループの間の処理を繰り返します。

文法 FOR i＝e TO m
［STEP s］
～
NEXT

i→ループ変数／e→初期値／m→終了値／s→増分（デフォルト値は1。STEPsを省略した場合は，STEP 1として，実行されます。）

説明 ループ変数は，カウンタとして使われ，最初に初期値に設定されます。そしてFOR文以降からNEXT文までが実行され，カウンタの値はSTEPによって指定された増分だけ増減されます。次にカウンタの値が終値と比べられ，カウンタの値が終値に達していなければ，FOR文の次の文へ戻り，同じ処理が繰り返されます。
STEP文が省略された場合，増分は＋1とみなされます。
また増分は，負の値をとることもできます。
FOR―NEXT は「入れ子構造」にすることができます。これは1つのFOR―NEXT

2	プログラムモード
	行番号（0～65532）をつけて文を入力した場合，それはメモリ中にプログラムとして行番号とともに格納（かくのう）されます。そして，いったん格納されたプログラムは，RUN コマンドおよび GOTO，GOSUB によって実行させられます。

「ダイレクトモード」は，プログラムをメモリの中に覚え込（こ）みません。プログラムを入力したら，すぐに実行します。

ですから，「ちょっと，プログラムを動かしてみたいな」という時，**使うと便利。**

みなさんにも，ダイレクトモードを使ってもらいたいと，思います。

次のように，キーボードから入力してみてくださいね。

```
SPRITE ON [RETURN]
FOR I=1 TO 10:DEF MOVE(0)=SP
RITE(RND(16),RND(9),1,100,1,
2):MOVE 0:PAUSE 80:NEXT [RETURN]
```

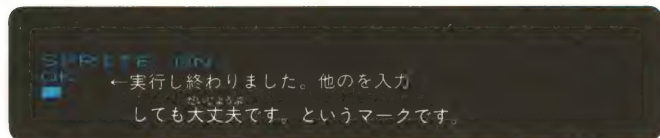
画面は

このプログラムを実行すると，いろんなキャラクタが，画面から外へ向って**飛び出して行きます。**

「ダイレクトモード」なので，

```
SPRITE ON [RETURN]
```

と入力すると，すぐに実行してしまいます。

SPRITE ON
OK ←実行し終わりました。他のを入力
しても大丈夫です。というマークです。

でも，**「プログラムモード」**の時は，ちょっと違うみたいです。

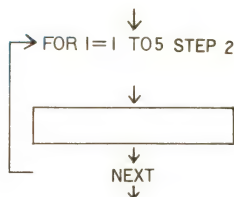
今度は，次のようなプログラムを入力してください。お願いします。

```
10 SPRITE ON [RETURN]
20 FOR I=1 TO 10:DEF MOVE(0)
=SPRITE(RND(16),RND(9),1,100
,1,2):MOVE 0:PAUSE 80:NEXT [RETURN]
```

プログラムの説明

10	スプライトを表示可能にします
スプライトを表示可能にします	

20	いろいろなキャラクタが飛び出す
キャラクタの種類，方向を乱数で決めます。そして，表示します。 これを，10回繰り返します。	

の中にもう1つのFOR—NEXTを置くことができるということです。1つのFOR—NEXTは，完全に他のFOR—NEXTの内部になければなりません。

● SPRITE ON

働き スプライト面の表示を可能にします。スプライト表示モード。

文法 SPRITE ON

説明 スプライト面をバックグラウンド面に重ねて表示できる状態にします。スプライト面上にあるスプライト（アニメキャラクタ）が，画面上に表示（見えるように）します。

DEF SPRITEまたはDEF MOVEを実行する前には，必ずSPRITE ONが必要です。SPRITE OFFが実行されるまで，継続して有効です。

● DEF MOVE

働き アニメキャラクタに対して固有の動きを定義します。

文法 DEF MOVE (n)＝SPRITE (A, B, C, D, E, F)

n→アニメキャラクタの動作番号（0～7）／A→アニメキャラクタの種類（0～15）／B→動作の方向指

定(０～８)／Ｃ→動かす速さ （１～255)／Ｄ→全移動量 （１～255)／Ｅ→表示優先度(０～１)／Ｆ→配色番号 （０～３）
説　明 16種類のアニメキャラクタの中から８種類を選び，その動きを定めます。

● RND

働　き 引数未満の乱数（整数）を発生します。

文　法 RND （X）
X→数式（整数） １～32767

説　明 発生した整数の乱数が，この関数の値になります。この乱数は「引数－１」までの乱数になっています。
RND （１） は 常に０です。

● MOVE

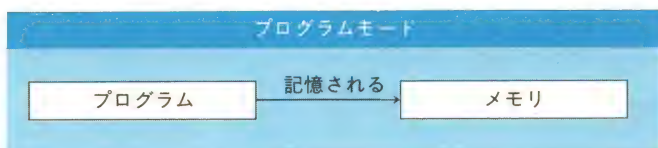
働　き アニメキャラクタの動きを開始させます。

文　法 MOVE n_0 [, n_1, …, n_7] n_0～n_7→DEF MOVE で定義した動作番号（０～７）

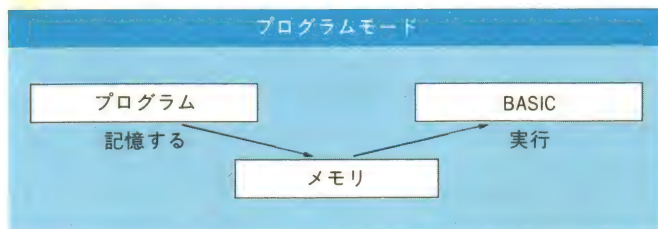
説　明 あらかじめDEF MOVE命令で定義されたアニメキャラクタの動きを開始させます。(SPRITE ON でスプライトを表示するようにしておいてください)
アニメキャラクタの動きを同時に８個まで開始させることができます。
MOVE文が実行されると，DEF MOVE命令で指定された動きが完了するまで，BASICのコントロールとは非周期にアニメキャラクタは動きます。つまり，プログラムが終了している場合でも，アニメキャラクタは，

でも，「プログラムモード」の場合，プログラムを入力しても，すぐ実行したりしません。
プログラムは，メモリに記憶(きおく)されます。

プログラムモード

プログラム ―記憶される→ メモリ

この記憶されている「プログラム」を実行させたい時は，**RUN命令**を使います。

RUN RETURN

「RUN命令」が入力されると，BASICは，メモリに記憶しておいた「プログラム」を読み出して，実行します。

構造について、です

「プログラム」は

DEF　MOVE 文で定義された移動量，方向指定，速度に応じ，動きが完了するまで動きます。

● PAUSE

働　き　プログラムの実行を一時停止します。

文　法　PAUSE　[n]

n → 0 ～32787

説　明　このステートメント（命令）を実行すると，設定単位時間分プログラムの実行を停止します。n を省略すると，何かのキー入力があるまでプログラムの実行を停止します。

● 行

行は BASIC プログラムの基本的な単位です。BASIC の行の書式では次の通りです。

行番号　命令 [：命令…]
〈キャリッジリターン〉

プログラムは1行に1つ以上のステートメント（命令）を入れることができます。それぞれのステートメントはコロン〈：〉で区切られている必要があります。BASIC の行は常に行番号ではじまり，キャリッジリターンで終わります。

●マリオ

というように，「行」が行番号の小さい順に並んでる，と説明しました。

で，実行する時も，やっぱり

行番号の小さい順に、実行

するそうです。

つまり，BASIC はプログラムの先頭から，順序よく「実行」してゆきます。

●レディ

プログラムの実行する順序

10	行
↓	
20	行
↓	
30	行
↓	
40	行
↓	
50	行
↓	
60	行

●ファイターフライ

●アキレス

例えば，次のプログラムを実行してみたい，と思います。

このプログラムを実行すると，**マリオ***，**レディ***，**ファイターフライ***，**アキレス*** が　順番に，画面の中央から現れます。

```
LIST ←―メモリに書きこまれているプログラムを表示します
10 DEF MOVE(0)=SPRITE(0,1,1,
125,1,0):MOVE 0:PAUSE 30
20 DEF MOVE(0)=SPRITE(1,3,1,
125,1,2):MOVE 0:PAUSE 30
30 DEF MOVE(0)=SPRITE(2,5,1,
125,1,3):MOVE 0:PAUSE 30
40 DEF MOVE(0)=SPRITE(3,7,1,
125,1,2):MOVE 0:PAUSE 30
OK
```

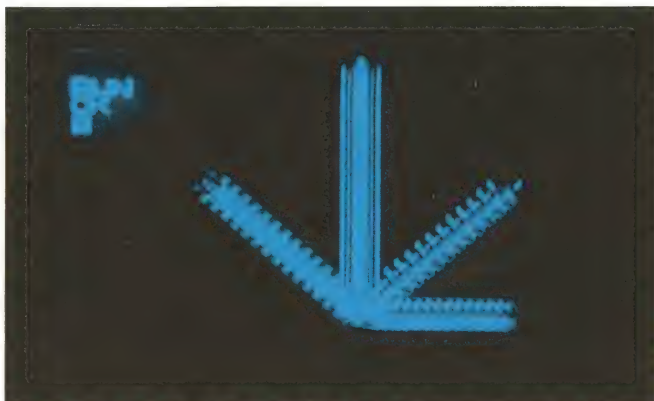
上の「プログラムの説明」を，見てくださいね。
そうすると，プログラムも

という順番に，並んでいます。

やっぱり，プログラムは，行番号の小さい順に実行されてゆくみたい，です。

実行して，確(たし)かめてみます…ね。

```
RUN RETURN
```

●無条件飛越し命令
無条件飛越し命令（unconditional Jamp instruction）は，無条件分岐命令（unconditional branch instruction）ともいう。
プログラムの他の部分に制御を移す命令。この移動はそのプログラムのそれまでの条件や状態には無関係である。
条件付き分岐命令（conditional branch）と対比される。

ジャンプは，無条件です

プログラムは

行番号の小さい順に実行

されます。

でも，これだけでは，**とても不便**なんだそうです。

そこで，この前の

	プログラムの構造の種類
1	ジャンプ（GOTO）
2	分岐（IF～THEN…）
3	繰り返し（FOR～NEXT）
4	サブルーチン（GOSUB, RETURN）

という，４つの構造を使います。

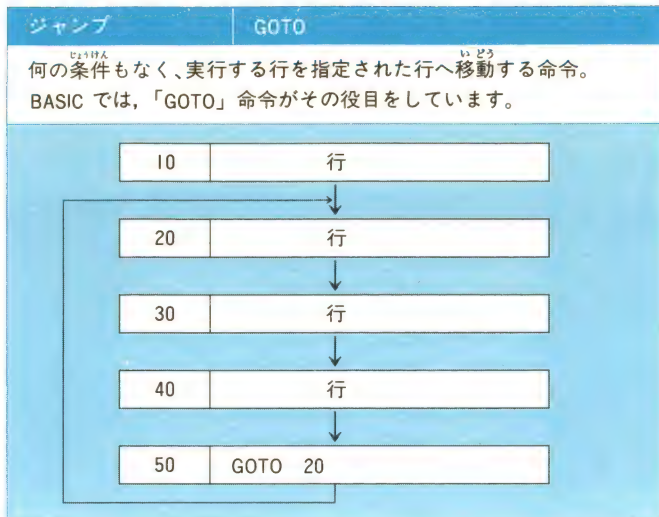
まず

ジャンプ(GOTO)

から，説明したいと思います。ジャンプは「無条件飛越し命令*」

とも言います。
　この「ジャンプ」は

ジャンプ	GOTO

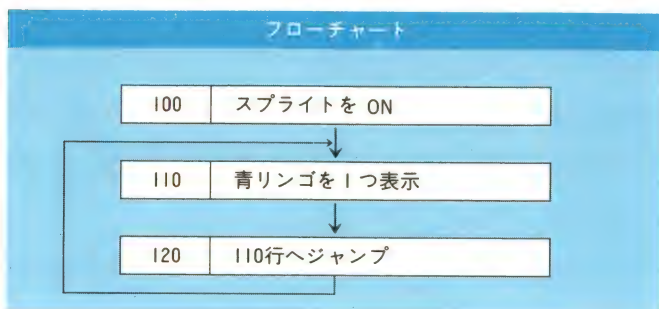
何の条件もなく、実行する行を指定された行へ移動する命令。
BASIC では，「GOTO」命令がその役目をしています。

という命令のことだそうです。
　BASIC では，GOTO 命令がこの役目をしています。

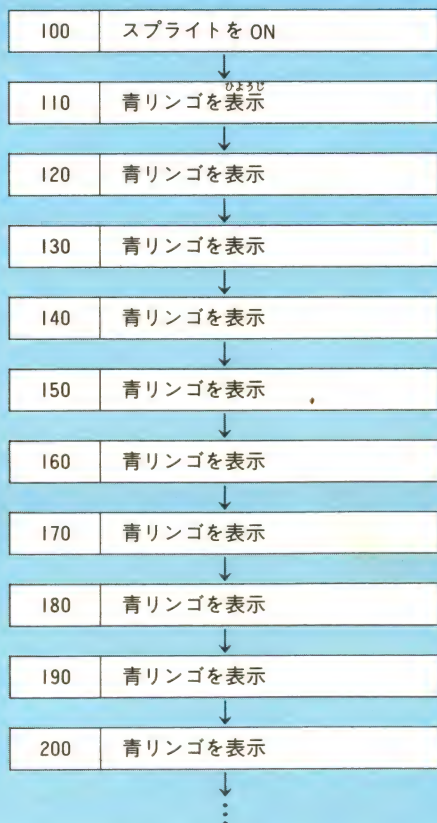
GOTO	無条件分岐命令

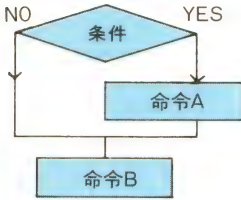
働き
指定した行番号へ無条件にジャンプします。
文法
GOTO ｛行番号｝
行番号→ジャンプ先の行番号
説明
GOTO 命令は指定した行番号に，無条件にジャンプするステートメント（命令）で，プログラムの流れを変えるときに使用します。ダイレクト命令では，変数の値をクリアせずに，特定行番号からプログラムを実行させることができます。

　この「GOTO」命令を使うと，今までにできなかったこと，できます。
　例えば **無限回**，繰り返したり…。同じことを，いつまでも，ず～っと繰り返します。

下のプログラムは，**青いリンゴ**をずっと表示します。止めたい時は STOP キーを押してくださいね。

```
100 SPRITE ON
110 PRINT CHR$(215);
120 GOTO 110
```

たった3行のプログラムなのに，とても沢山のことをします。
でも。
これを，GOTO命令を使わないで，作ったら…。

```
100 SPRITE ON
110 PRINT CHR$(215);
120 PRINT CHR$(215);
130 PRINT CHR$(215);
140 PRINT CHR$(215);
150 PRINT CHR$(215);
160 PRINT CHR$(215);
170 PRINT CHR$(215);
180 PRINT CHR$(215);
190 PRINT CHR$(215);
200 PRINT CHR$(215);
        :
```

●条件付き分岐命令

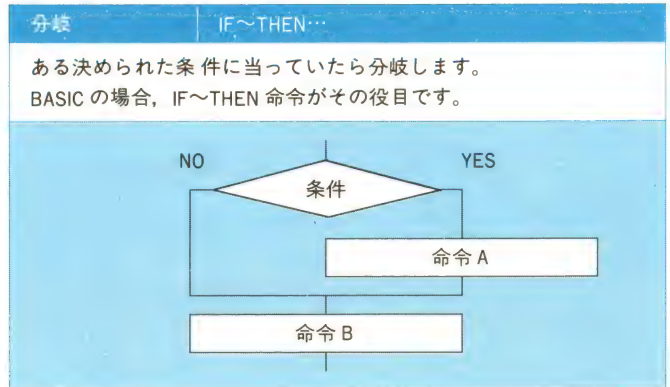
ある特定の条件が満たされた時，たとえば指定されたデータ項目がゼロの場合にのみ，制御をプログラムの他の部分に移す分岐命令。特定の条件を満たさない場合，次の命令に進む。

もしも，○○だったら…

今度は

分岐(IF～THEN…)

について，説明します。分岐(ぶんき)は，条件(じょうけん)付き分岐命令*とも言うそうです。

この「分岐」命令は

分岐	IF～THEN…
ある決められた条件に当っていたら分岐します。 BASIC の場合，IF～THEN 命令がその役目です。	

というものです。

日本語になおすと

もしも○○が××だったら△△

ということです。

そうですね…。例えば　こんなふうに使います。

これは，雨が降りそう…。そんな時のわたしの学校へ行く様子です。もちろん，お友だちを誘ったり，ということは書きませんでした。

BASIC では，**IF～THEN…命令**がこの役目をしています。

関係演算子

関係演算子は２つの数値や文字列を比較するときに使います。結果は，真(正しい)，偽(正しくない)で得られ，条件付き分岐命令(IFなど) で，プログラムの流れを変えるのに使われます。

例

IF　X＝0　THEN　1000

(もしも　変数Xが0ならば　1000行へジャンプ)

IF　A$＝〝Y〟　THEN　PRINT　〝YES〟

(もしも　変数A$の内容が〝Y〟ならば　〝YES〟を表示)

関係演算子	意味
＝	両辺が等しい
＜＞	両辺が等しくない
＜	左辺が右辺より小さい
＞	左辺が右辺より大きい
＞＝	左辺が右辺より大きいか等しい
＜＝	左辺が右辺より小さいか等しい

論理演算子

論理演算子は，複数の条件を調べたり，ビット操作やブール演算を行ったりするのに使われます。

例

IF　X＝0　AND　Y＝0　THEN　250

(もしも　変数Xが0でしかも変数Yが0ならば　250行へジャンプする)

IF　A$＝〝A〟　OR　A$＝〝B〟　THEN　C＝12

(もしも　変数A$が〝A〟または変数A$が〝B〟ならば変数Cに12を代入する)

演算順位	論理演算子	意味
1	NOT	否定
2	AND	論理積
3	OR	論理和
4	XOR	排他的論理和

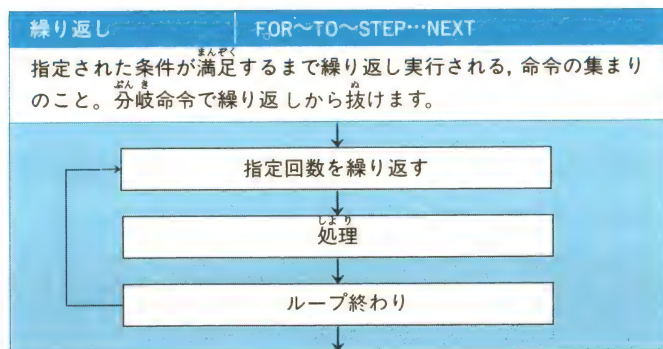
FOR，NEXT しましょ

次は

繰り返し

について，説明したいと思います。繰り返しは「ループ*」とも呼ばれるそうです。

この「繰り返し」構造は，いくつかの命令の集まりを，**何回も繰り返して実行したい時**，使います。

● ループ（loop）
指定された条件が満足されるまで繰り返し実行される一連の命令・分岐命令でループから出る。

「ジャンプ」構造で，無限に繰り返すこと，できました。

でも，無限は，困っちゃいます…。

そんな時，この「繰り返し」構造を使います。BASICでは，**FOR…NEXT命令**が，この役目。

● フローチャート

フローチャート（flow chart）は，流れ図ともいう。

事象の系列の図式的表現。普通，種々の事象とそれらの結合を示す約束された記号を用いて書く。

● プログラム・フローチャート

デジタル計算機プログラムの一連のステップの論理的関係を図式表現するのが目的。プログラムのフローチャートを書くには，種々のレベルの複雑さが考えられるが，少なくとも概略的な流れ図と詳細な流れ図の2つが普通用意される。

● DEF SPRITE

働　き　スプライト画面上に表示したいスプライト（アニメキャラクタ）を定義します。

文　法　DEF SPRITE n,（A, B, C, D, E）＝文字式

文字式→CHR$(N)，または文字例〝@ABC〟または文字変数／→スプライト番号（0〜7）／A→配色番号（0〜3）／B→キャラクタ合成型（0〜1）／C→表示優先度（0〜1）／D→X軸方向反転指示（0〜1）／E→Y軸方向反転指示（0〜1）

説　明　スプライト面に表示したいアニメキャラクタを定義します。

スプライト番号は，0〜7の8つまで指定し，定義できます。

キャラクタ合成型に0を指定すると，1×1キャラクタ（タテ1文字ヨコ1文字）でアニメキャラクタを構成

FOR〜TO〜STEP…NEXT　繰り返し

働　き

FOR—NEXT グループの間の処理を繰り返し実行します。

文　法

```
FOR  i=e  TO  m 〔STEP  s〕
 ∫
NEXT
```

i →ループ変数

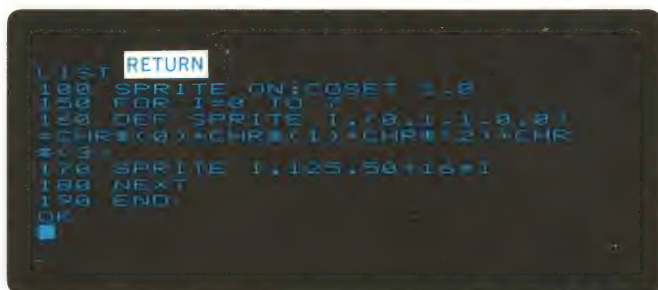
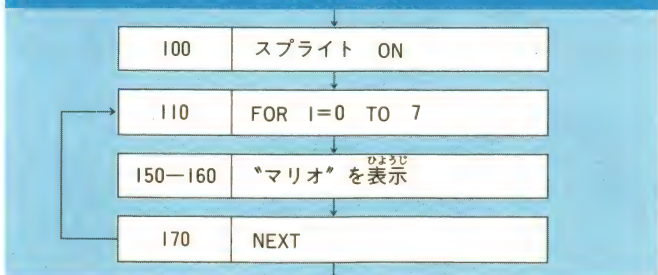
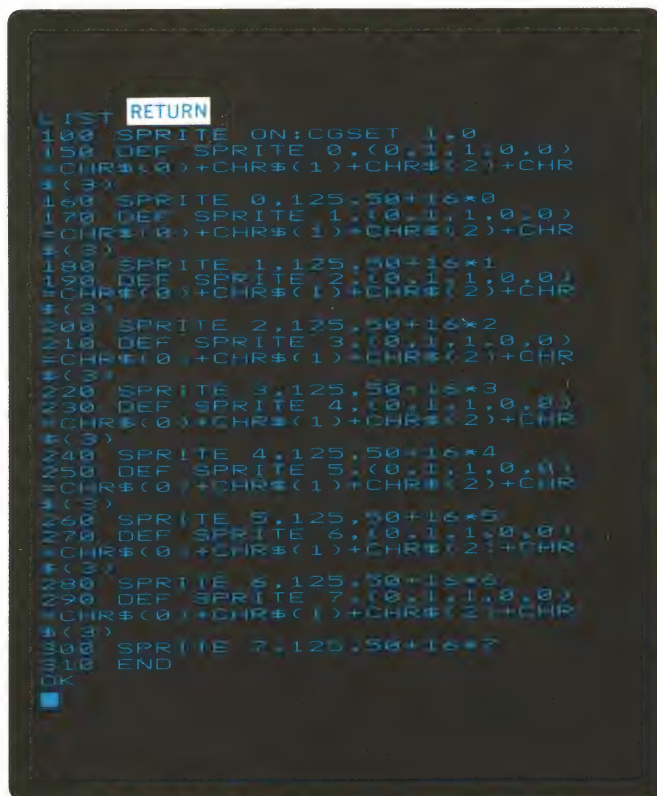
e →初期値

m→終了値

s →増分

説　明

FOR—NEXT グループにおいて，FOR 文はループの先頭，NEXT 文はループの終わりを示しています。STEP s を省略すると増分は1になります。ループ変数 i の初期値 e と終了値 m の条件がすでに満たされている場合は，そのまま NEXT 文の次の行へ移ります。

この「FOR—NEXT」命令を使って，**〝マリオ〟をタテに8人並べるプログラム** を作ってみました。

でも。同じプログラムを，FOR−NEXT 命令を使わないで，作ったら…。こんなふうに，なってしまいます。

します。 1 を指定すると，2×2キャラクタ（タテ2文字ヨコ2文字）でアニメキャラクタを構成します。
表示優先度に0を指定すると，バックグラウンド面よりも前に，1を指定すると後に，アニメキャラクタを表示します。
X軸方向・Y軸方向の反転指示に1を指定すると，それぞれの軸に対して反転したキャラクタで定義されます。

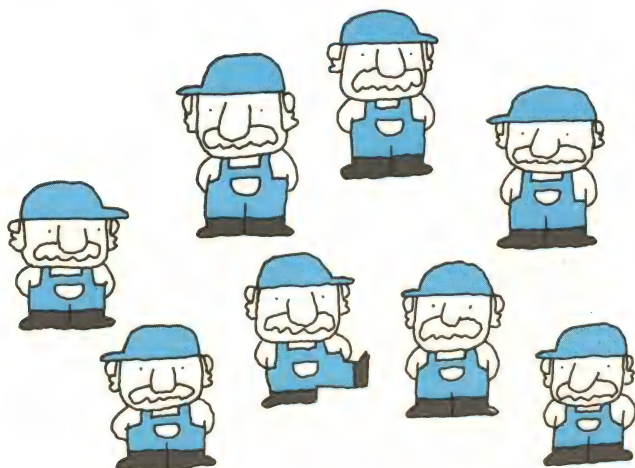
● SPRITE

働　き 定義されているスプライトを任意の位置に表示します。

文　法 SPRITE n [,X, Y]
n→スプライト番号（0〜n）/X→スプライト面水平方向の座標（0〜255）／スプライト面垂直方向の座標（0〜255）

説　明 すでにDEF SPRITE命令で定義されているアニメキャラクタをスプライト面上の任意の位置に表示します。
水平方向，垂直方向の座標を省略すると，表示されているアニメキャラクタの中から，指定のスプライト番号のアニメキャラクタのみを消去します。

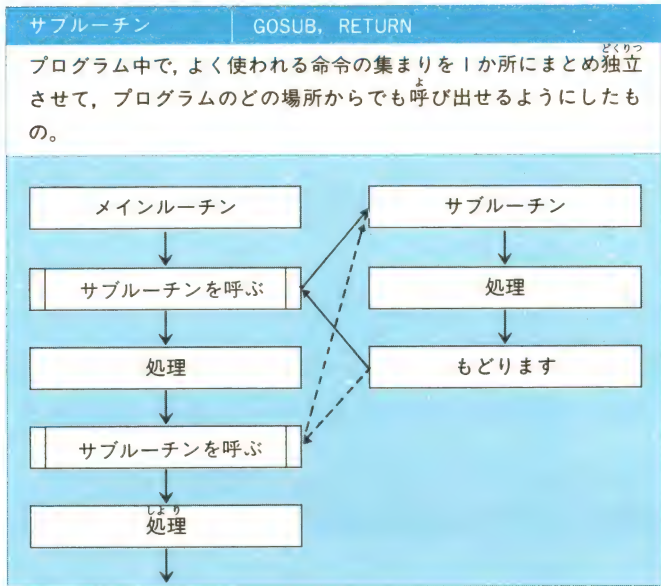
フローチャート

100	スプライトON。
150-160	マリオを表示(ひょうじ)します
170-180	マリオを表示します
190-200	マリオを表示します
210-220	マリオを表示します
230-240	マリオを表示します
250-260	マリオを表示します
270-280	マリオを表示します
290-300	マリオを表示します
310	終わり

必要なものは，まとめます

さて，プログラムの構造の４番目の

サブルーチン(GOSUB,RETURN)

について，説明したいと思います。

サブルーチンは

という構造のことだそうです。

BASICでは，**GOSUB命令とRETURN命令**が，この役目をしています。

GOSUB	サブルーチンを呼ぶ命令

働　き

指定したプログラム内のサブルーチンを呼び出します。

文　法

GOSUB｛行番号｝

行番号 → サブルーチンの開始行番号

● メインルーチン

メインルーチン（main routine）は，メインプログラム，主プログラムともいう，プログラムの中心的な部分ですって。処理されるデータの性質やプログラムの演算中に起こる条件に従ってサブルーチンへ制御を移す。いろいろなサブルーチンを管理する。中心的な枠組み。

● サブルーチン（subroutine）

プログラムの全体の機能のうち論理的にまとまった一部を遂行し，それを構成する特定の命令群が必要な時はいつでも利用できるようになっているプログラムの一部分。

● ループ（loop）

指定された条件が満足されるまで繰り返し実行される一連の命令。

● サブルーチン（subroutine）

プログラムの機能のうち論理的にまとまった一部を遂行し，それを構成する特定の命令（instruction）群が必要な時はいつでも利用できるようになっているプログラムの一部分。
サブルーチンを構成している命令群は，必要なだけ繰り返し存在する必要はなく，主プログラムから分岐によってはいることができる。

説　明

IF 命令（分岐構造）やFOR-NEXT命令（繰り返し構造）で作ったループ*は，同じ処理を何回も続けて行うものでしたが，プログラムの所々で同じ処理を行う必要がある場合には，何回も繰り返して使う部分をサブプログラムとして組んでおき，必要に応じてそれをプログラム本体（メインプログラム）から呼び出すようにします。メインプログラムから呼び出されるサブプログラムのことをサブルーチンといい，このサブルーチンを呼び出す命令として，GOSUB が用意されています。
GOSUB のあとについている行番号はサブルーチンの開始行で，呼び出されるサブルーチンの最後には必ず RETURN をつけます。サブルーチンの中からさらに別の GOSUB 命令を使ってもかまいません。

RETURN　サブルーチンから戻る命令

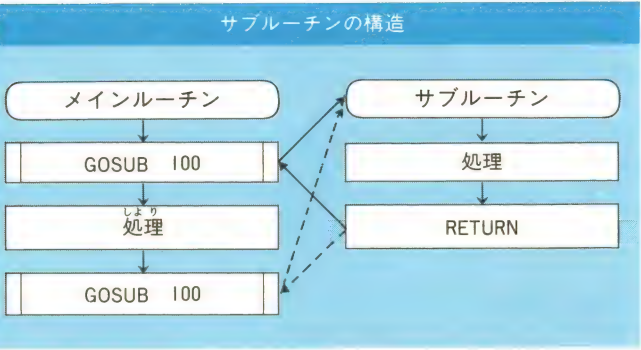
働　き

サブルーチンから，呼び出したルーチンへ戻ります。

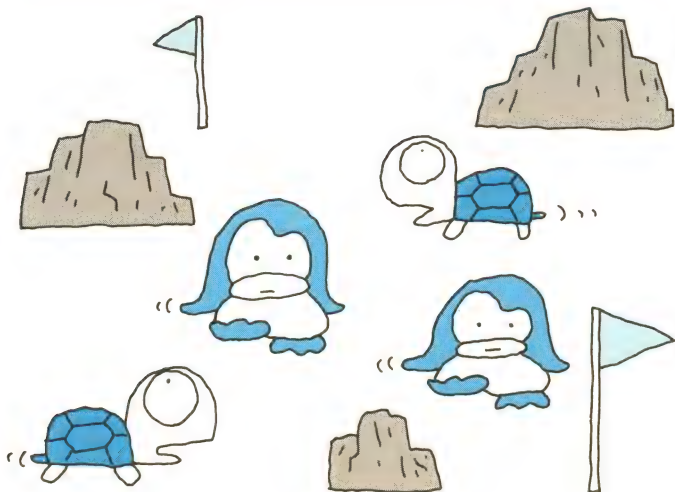
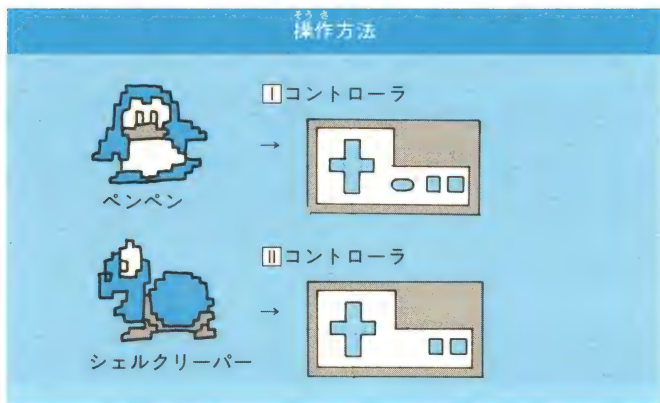
文　法

RETURN〔{行番号}〕

説　明

RETURN は，GOSUB で呼ばれたサブルーチンの最後につけて，呼び出したルーチンへ戻るときに使います。
行番号を指定するとその行に戻りますが，普通，行番号を省略して使います。省略すると GOSUB 命令の後ろに戻ります。

例えば，次のような「サブルーチン構造」を使っていないプログラム。

このプログラムは，Iコントローラと IIコントローラを使って，アニメキャラクタを動かすものです。Iコントローラで **「ペンペン」** を，IIコントローラで **「シェルクリーパー」** を動かします。

でも。ゲームじゃないです。 **「ちょっとしたミニ・プログラム」** です。

● VIEW（ビュー）

働　き BG GRAPHIC 面をバックグラウンド面へコピー（複製）します。

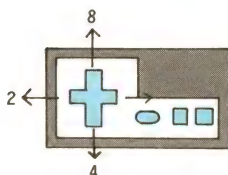
文　法 VIEW

説　明 BASIC を実行中，VIEW コマンド（命令）を入力すると，BG GRAPHIC 面に描いた絵がバックグラウンド面へコピー（複製）します。バックグラウンド面にコピーされた BG GRAPHIC は BASIC で書き換えができますが，BG GRAPHIC 面にはその絵が残っています。
BG GRAPHIC 面に描いた絵をそのままの色で，バックグラウンド面にコピー表示するためには，バックグラウンド面のパレットコードは 1（CGSET 1，1）を使用してください。

● STICK（スティック）

働　き コントローラのボタンからの入力値を与えます。

文　法 STICK（X）
X → 0，1（コントローラの番号）

説　明 コントローラのボタンの値を与えます。

プログラムの説明

100-110	画面の設定をします
	スプライトをONにして，バックグラウンド用パレットコードを1，スプライト用パレットコードを2にします。それから，BG GRAPHIC面をバックグラウンド面にコピーします。

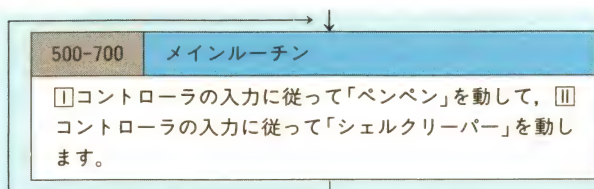
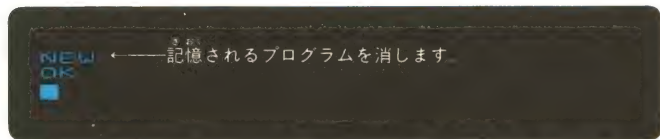
↓

200-230	スプライトを設定します
	200行は注釈です。スプライト0に「ペンペン」を，スプライト1に「シェルクリーパー」を設定します。

↓

300-320	変数を設定します
	変数X（ペンペンのヨコ座標）←100 変数Y（ペンペンのタテ座標）←125 変数M（シェルクリーパーのヨコ座標）←150 変数N（シェルクリーパーのタテ座標）←125

このプログラムを入力する前に

NEW RETURN

と，ダイレクトモード*で入力します。こうすると，メモリの中に記憶されてたプログラム，すべて消えてなくなります。

次は，**「BG GRAPHIC」**で，ゲームの背景を描きます。

これは，

「背景があった方が，面白いな」

と，思う人だけで，いいです。このプログラムは，背景がなくっても，動きます。でも，ちょっぴり，さみしい…です。

BG GRAPHIC で背景を描く人は,本の終わりにある**「BG GRAPHIC画面表」**を見て，描いてくださいね。BG

●ダイレクトモード

行番号をつけずにBASICの文法に沿った文を入力した場合，その文は，キャリッジリターンを入力後，すぐに実行されます。これをダイレクトモードにおける実行といいます。

●プログラムモード

行番号（1～65535）をつけて文を入力した場合，それはメモリの中にプログラムとして格納されます。そしていったん格納されたプログラムは，RUN 命令および GOTO 文，GOSUB 文によって実行できます。これをプログラムモードにおける実行といいます。

●文

文とは，BASIC が実行する手続きを書いた最小単位のことである。

文には，BACIC が実行する式，ステートメント，コマンド，関数などを書くことができます。

●RUN（ラン）

働き　プログラムを実行させます。

文法　RUN　n

n→実行を開始させたい行の行番号

説明　RUN は，プログラムを実行させるための命令です。RUN を入力すると，プログラムは最初から順に実行されますが，このとき変数はすべてクリアされます。また，プログラムを途中から実行させたい場合は，実行開始行番号 n をつけます。

GRAPHIC 画面表の中の，**「ペンペンと仲間(なかま)たち」**が，このプログラムの名前です。

そして　プログラムを入力して，**RUN命令**で実行すると，次のような画面に，なります。

宇宙(うちゅう)も寝(ね)返る，サブルーチン

実は。

このプログラムの中に，**同じ命令の集まりが2か所**あるんです。

もう，気がついていたかも…。ううん，たぶん，みなさんは，とっくに気がついてる，と思います。

もう一度，プログラムを見ると――。

```
LIST [RETURN]
```

```
="****"
220 DEF SPRITE 1,(0,1,0,0,0)
=CHR$(184)+CHR$(185)+CHR$(18
6)+CHR$(187)
300 ' * ヘンスウ ノ セッテイ *
310 X=100:Y=125
320 M=150:N=125
500 ' * メイン ルーチン *
510 I=STICK(0)
520 VX=0:VY=0
530 IF I=1 THEN VX= 1:VY=0
540 IF I=4 THEN VY= 1:VX=0
550 IF I=2 THEN VX=-1:VY=0
560 IF I=8 THEN VY=-1:VX=0
570 X=X+VX:Y=Y+VY
580 SPRITE 0,X,Y
590
600 I=STICK(1)
610 VX=0:VY=0
620 IF I=1 THEN VX= 1:VY=0
630 IF I=4 THEN VY= 1:VX=0
640 IF I=2 THEN VX=-1:VY=0
650 IF I=8 THEN VY=-1:VX=0
660 M=M+VX:N=N+VY
700 GOTO 500
OK
```

●バイト（byte）
一つの単位として考えられる２進数字の集まり。

やっぱり，同じ「命令の集まり」が，２か所ありました。
こういう

同じ処理はサブルーチンにする

方が良い，そうです。

サブルーチンにすると，**メモリの節約になる，プログラムがきれいになる**などの「特典」があります。特に，メモリの節約になるのは，とっても嬉しいです。

ファミリーベーシック V2.0の RAM の量は，**2Kバイト**です。「２Ｋバイト」は，2000文字くらい記憶できる，ということです。

ちょっと，少ない雰囲気…。

なので，メモリの節約になるのは，**宇宙がひっくりかえっちゃう**ほど，嬉しいんです。

	サブルーチンにすると　こんなにお得です
1	メモリの節約になります
2	プログラムがきれいになります

サブルーチンにするには，次のように入力します。

```
800 ' * サブ゛ルーチン * RETURN
810 VX=0:VY=0 RETURN
820 IF I=1 THEN VX= 1:VY=0 RETURN
830 IF I=4 THEN VY= 1:VX=0 RETURN
840 IF I=2 THEN VX=-1:VY=0 RETURN
850 IF I=8 THEN VY=-1:VX=0 RETURN
860 RETURN RETURN
520 GOSUB 800 RETURN
530 RETURN
540 RETURN
550 RETURN
560 RETURN
610 GOSUB 800 RETURN
620 RETURN
630 RETURN
640 RETURN
650 RETURN
```

すると。

いつの間にか，プログラムは，**サブルーチン化**されてしまいました。

```
LIST RETURN
100 SPRITE ON:CGSET 1,2
110 VIEW
200 ' * スプライト ヲ セッテイ *
210 DEF SPRITE 0,(0,1,0,0,0)
="[illegible]"
220 DEF SPRITE 1,(0,1,0,0,0)
=CHR$(184)+CHR$(185)+CHR$(18
6)+CHR$(187)
300 ' * シンスウ ヲ セッテイ *
310 X=100:Y=125
320 M=150:N=125
500 ' * メイン ルーチン *
510 I=STICK(0)
520 GOSUB 800
570 X=X+VX:Y=Y+VY
580 SPRITE 0,X,Y
590
600 I=STICK(1)
610 GOSUB 800
660 M=M+VX:N=N+VY
670 SPRITE 1,M,N
700 GOTO 500
800 ' * サブ゛ルーチン *
810 VX=0:VY=0
820 IF I=1 THEN VX= 1:VY=0
830 IF I=4 THEN VY= 1:VX=0
840 IF I=2 THEN VX=-1:VY=0
850 IF I=8 THEN VY=-1:VX=0
860 RETURN
OK
```

フローチャートの変化
メインルーチン
Ⅰコントローラを調べます
動く方向を決めます
ペンペンを表示します
Ⅱコントローラを調べます
動く方向を決めます
カメさんを表示します
サブルーチン
メインルーチン
サブルーチン
Ⅰコントローラを調べます
動く方向を決めます
GOSUB 800
ペンペンを表示します
Ⅱコントローラを調べます
GOSUB 800
カメさんを調べます

変数，です！

ふうッ。疲れちゃった！
この前のところで

プログラムの構造

について，みなさんに，知ってもらいました。
ちょっと，難しかったと思います。わたし，せいいっぱい**一生懸命**，説明しました…。
「ん？　ぜーんぜん難しくなかったよ」
なんて，人。わたし，尊敬しちゃいます。

勇気を出して，告白します。

わたくし，この「プログラムの構造」は，はじめ，ぜんぜん分りませんでした。何だか，さっぱり…。
でも。
いろんな命令を試したり，うんと短かいミニ・プログラムを作ったりしているうちに，なんとなく分ってきたんですよ。段々と，ゆっくりと，霧が晴れてゆくみたいに…。
なので，よく分らなくても，安心してくださいね。
プログラムを作る時。
なるべくなら，**フローチャート***を書いてから，作った方が，いいみたいです。詳しく書かなくても，こんなようなプログラム作ろうかな，なんて思ったら，図にしてみると，**プログラムの構造が分ってくる**と思います。
それでは

変数

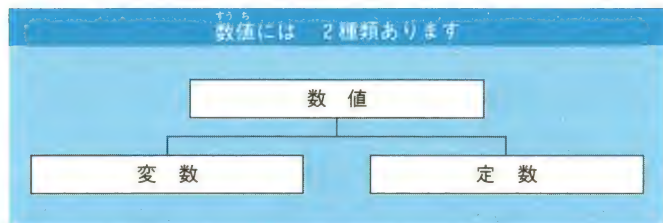
について，説明したいと思います。

●フローチャート
事象の系列の図式的表現。普通，種々の事象とそれらの結合を示す約束された記号を用いて書く。
本文中の場合，プログラムフローチャートを指す。

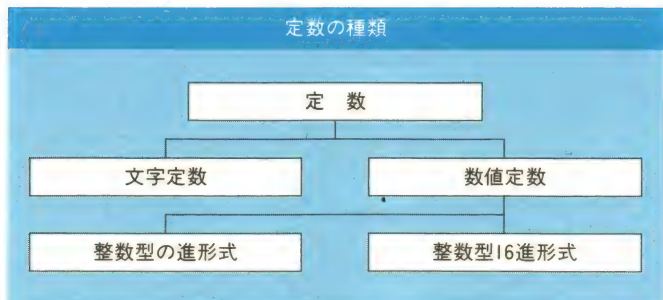
●プログラム・フローチャート
デジタル計算機プログラムの一連のステップ間の論理関係を図式表現するのが目的。プログラムのフローチャートを書くには，種々のレベルの複雑さが考えられるが，少なくとも概略的な流れ図と詳細な流れ図の2つが，普通，用意される。

●変数（variable）
プログラムの進行中に変わりうるデータが変数である。
変数はまた，記憶装置における データの記憶場所であると考えてもよい。
記憶場所の内容が変数の値である。また，記憶場所の名前が，すなわち変数名である。

定数は，変えられません

「NS-HuBASIC*」の扱える数の値には**定数***と**変数**の2種類がある，そうです。

名　称	意　味
定　数	値は一定で，変わることがありません。
変　数	プログラムの指示により値を変えます。

「定数」は，プログラムでどんなことをしても，変わらない数値のこと。定数はその値を変えることができません。

この定数には，**文字定数，数値定数**の2種類があります。数値定数は，**整数型10進形式***と**整数型16進形式***に，分けることができます。

そして，文字定数は，**31文字までの文字の所**のこと。数値定数は，**正または負の整数**のことです。

● NS-HuBASIC
NS-HuBASIC（エヌエス・ヒューベーシック）は，ファミリーコンピュータ用プログラム言語である。開発は，ハドソンが中心となり任天堂，シャープが協力して行われたと考えられている。

● 定数（constant）
値の一定なデータが定数である。定数はふつう数学で用いられているのと同じ意味であるが，例えば1.23E－4が1.23×10^{-4}であるという，指数部を表わす特別の規約がある。

● 整数（integer）
小数を含まない数のこと。

● 10進形式
10進数（decimal numeral）は，10進法で示される数のこと。すなわち，0，1，2，3，4，5，6，7，8，9という10文字を使って表わされている数。

● 16進形式
16を底あるいは基数とする数表現。0から9の10個の数字の他に，さらに6個，通常，A，B，C，D，E，Fが，それぞれ10，11，12，13，14，15，を1文字で表わすのに使われる。

文字定数

文字定数は，31文字以下のダブルクォーテーション〈"〉で囲まれた，英数字，カナ文字，記号などの列のことです。

例　"1 2 3 4 5 6 7 8 9 0"
　　"キムラカナエ＆サイトウチアキ"
　　"GOOD　MORNING!!"

数値定数

数値定数は，正か負の数，または0のことです。

(1)　整数型10進形式

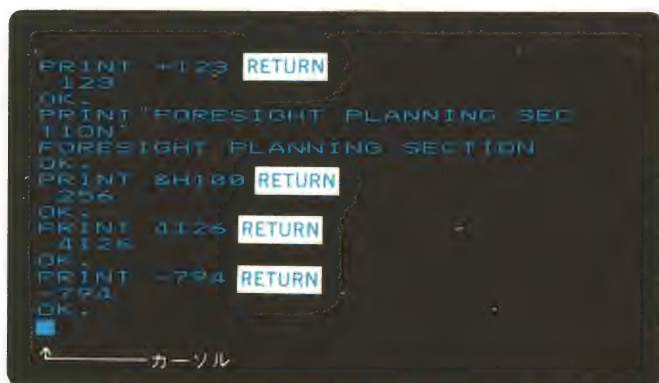
−32768〜＋32767までのすべての整数。小数点を付けることはできません。負の数の時は，かならず符号〈−〉を付けなくてはいけませんが，正の数の符号は〈＋〉を付けないこともできます。

例　32767
　　−123

(2)　整数型16進形式

&H0000〜&HFFFF までのすべての16進整数。PRINT 命令で，普通の10進数（整数のこと）になおして表示できます。

例　&HFFFF
　　&H100

この定数は，PRINT 命令を使って，画面に表示できます。
ちょっと「ダイレクトモード」で試してみますね。

でも，変数は変えられます

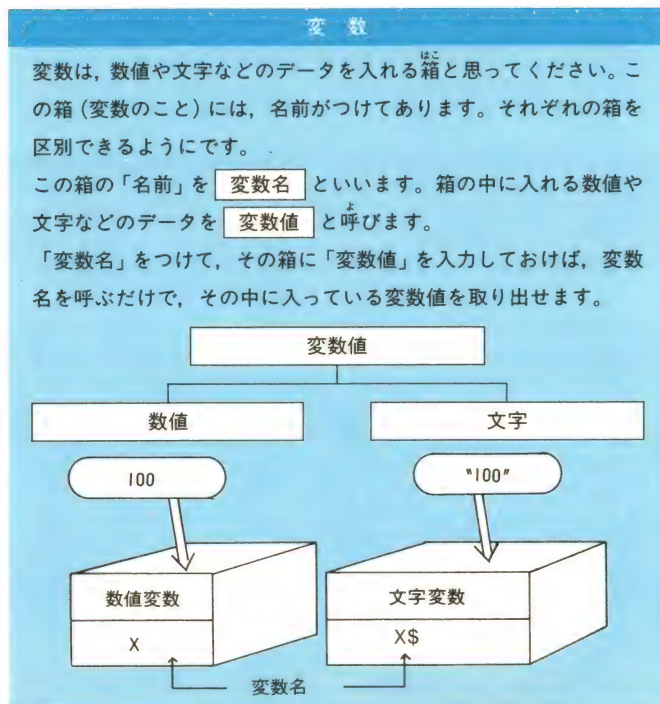
変数は，定数と違って，プログラムの指示で**値を変えられる数値**です。

この変数には**文字変数**と**数値変数**の２種類があります。

変数

変数は，数値や文字などのデータを入れる箱と思ってください。この箱（変数のこと）には，名前がつけてあります。それぞれの箱を区別できるようにです。

この箱の「名前」を 変数名 といいます。箱の中に入れる数値や文字などのデータを 変数値 と呼びます。

「変数名」をつけて，その箱に「変数値」を入力しておけば，変数名を呼ぶだけで，その中に入っている変数値を取り出せます。

「数値変数」は。**数値**を入れておく変数です。

変数名には，普通，英文字２文字か，英文字１文字に数字１文字をつけます。

そして「文字変数」は，**文字列**を入れておくための変数です。変数名には　普通，英文２文字か，英文字に数字をつけた後に〈$〉をつけます。

数値変数

数値変数の変数名は，A や B などの記号で名前をつけます。変数名は255文字までつけられます。でも，コンピュータは最初の 2 文字だけで，変数名を区別します。
それから，最初の 1 文字目は，必ず英大文字を使います。ただし，英文字であっても BASIC の命令は使えません。
変数値には，整数を使います。

例
A
XX
A1
FORESIGHT

文字変数

文字変数の変数名は，A や B などの記号の後に〈$〉をつけて，数値変数と区別します。変数名は255文字までつけられます。でも，コンピュータは最初の 2 文字だけで，変数名を区別します。それから，最初の 1 文字は，必ず英大文字です。
変数値は，ダブルクォーテーション〈"〉で囲まれた型で入力します。
ただし，文字数の変数値は，数字でも，「数字の文字列」として扱われます。

例
A$
AB$
Q1$
KANAE$

大切な命令だから

ここまで読んできた，みなさんは，**プログラム入力**の方法，**キーボード練習**の方法，**プログラム**について，**プログラムの構造**について，**変数**について。

それから，それから…。

うーん，とにかく　とても沢山(たくさん)のこと，分かったと思います。

「プログラム」は，命令の集まり，でした。

「プログラムの構造」には　ジャンプ，分岐(ぶんき)，繰(く)り返し，サブルーチンの４種類が，ありました。

「変数」は，プログラムの指定によって変わる値のこと，でした。

プログラム

行番号	行

↓

行番号	行

↓

行番号	行

↓
⋮

	プログラム構造の種類
1	ジャンプ（GOTO）
2	分岐（IF～THEN…）
3	繰り返し（FOR～NEXT）
4	サブルーチン（GOSUB，RETURN）

	意　味
定数	値(あたい)は一定で，変わることはありません。
変数	プログラムの指示(しじ)によって，変わります。

こんなに，説明してたんですね。いつの間にか…。感心，感心。
今度は，最後の仕上げです。ここでは

大切な命令

について，説明しようかな，って思っています。

REM と PRINT と INPUT と…

えーと。
まずは　**REM文, 代入文, PRINT文, INPUT文, END文, NEW命令,** について説明します。

命令文	命令の意味
REM	注釈を入れます
PRINT	データを画面に表示します
INPUT	キーボードからデータを入力します
代入文	変数に式の値を代入します
END	実行を終了します
NEW	プログラムを消去します

この本の終わりにある「BG GRAPHIC 表」の中の，ハートですきして，の図を見て，背景を作ります。
だいたい，次の写真のような画面に，なるはずです。

今度は，プログラムを入力します。

このプログラムは，REM 文，代入文，PRINT 文，INPUT 文，END 文，NEW 命令を，ぜんぶ使っています。

プログラムの説明

100	注　釈
このプログラムのタイトルをプログラムの中に，注釈としていれました。	

200-220	画面設定と名前の入力
画面をスプライト ON にして，パレットを1,2にします。それから，あなたの名前を入れてもらいます。BG　GRAPIC 画面をバックグラウンドにコピーします。	

300-340	MOVE を設定
アニメキャラクターでマリオとレディにして，動きの方向と速度と距離を決めます。	

500-610	メインプログラム
あなたとわたしの名前を，ハートの中に書いて…。そして，マリオとレディが走ってきて，キスしてしまうんです。	

わたしの作った，大切なプログラム，です。

みなさん，入力してくました？　入力したら，RUN 命令を使って，実行します。

RUN RETURN

REMは，注釈(ちゅうしゃく)です

このプログラムを止めたい時は STOP キーを押(お)してくださいね。実行が止まって，いつでも入力できるようになります。

このプログラムを見ながら，**大切な命令**説明したいと思います。

もう一度，プログラムのリスト…。

```
LIST RETURN
```

```
LIST RETURN
100 REM**** [illegible] KISS!! ****
200 SPRITE ON:COSET 8,2
210 INPUT "[illegible]";Y$
220 VIEW
300 '* [illegible] *
310 DEF MOVE(0)=SPRITE(0,3,5
,32,0,1)
320 DEF MOVE(1)=SPRITE(1,7,5
,32,0,1)
330 POSITION 0,7,106
340 POSITION 1,229,106
500 '* [illegible] *
520 LOCATE 10,4:PRINT Y$
530 LOCATE 10,6:PRINT "[illegible]
"
550 MOVE 0,1
560 PAUSE 750:PLAY "O4T1AB"
600 LOCATE 11,15:PRINT "KISS!
!"
610 GOTO 610
Ok
```

まず，

REM文

です。

REM 文は**プログラムを読みやすくするための命令**です。

プログラムを読みやすくするのは，わたし達（たち）“人間”のため…。

BASICには，ぜんぜん関係ないです（かえって，わずらわしくなるそうです)。

なので，BASICは，REM文のある行を**無視**(むし)して次の行へ移(うつ)ってしまいます。

REM　注釈(ちゅうしゃく)

働き

プログラム文にコメントを入れます。

文法

REM〔コメント〕

コメント→任意(にんい)の文字列（メッセージ）255文字まで。GOSUB文のサブルーチンに使用する場合は〈'〉（アポストロフィ）を使用してください。

省略形(しょうりゃくけい)

'（アポストロフィ）

説明

プログラムにコメントを挿入(そうにゅう)するためのステートメント（命令）で，プログラムの実行にはかかわりありません。

REMは〈'〉アポストロフィで代用できます。

REM文で書かれたプログラム行は，何も実行されません。

（プログラム用メモリは使われます）

このプログラムでは，100行と300行と，それから500行に使いました。

100行には，このプログラムのタイトルを書いて。300行と500行には，そこではどんなことをしてる，のかを書きました。

ちょっぴりですけど，読みやすいプログラム・リストになった，と思います。

INPUT は, 入力です

それから，このプログラムでは，

INPUT文

も，使ってみたりしました。

INPUT文は，**キーボードから入力したデータを変数に入れる**ための命令だそうです。

例えば，マイコンに「星うらない」をしてもらう時，**わたしの生年月日、**マイコンに知ってもらいます。そんな時に，このINPUT文，使われます。

●データ（data）
何らかの条件，値，または状態を示す数字。アルファベット文字，または記号からなる演算数，または因子の集まりを記述するのに使用される一般的表現。

INPUT	文字入力

働き

キーボードから数値(すうち)や文字を入力します

文法

INPUT〔"文字列"〕{ ; , } 変数〔, 変数, …〕

文字列→画面に表示(ひょうじ)する文字列

変数→キーボードから入力する数値や文字が入る変数

説明

キーボードから入力した数値や文字のデータを変数に入れます。あるデータを入力する場合，前もってデータを入れる変数を用意しなければなりません。そのためINPUTのあとには，必ず変数をつけます。変数は，文字変数，数値変数どちらでもつけることができます。また，変数は〈,〉(カンマ)で区切れば，複数個(ふくすうこ)つけることができます。この場合はデータを入力するとき，対応(たいおう)するデータを〈,〉で区切って1度に入力します。入力するデータの数は，INPUTのあとにつけた変数の数と一致(いっち)していなければなりません。

INPUT文を実行すると，画面に「?」を出力して入力待ちの状態(じょうたい)となりますが，「?」だけでは何を入力すればいいのかわかりません。そこで，INPUTには文字列のPRINT機能(きのう)がついていて，画面に表示したい文字列を〈"〉で囲んでから〈;〉(セミコロン)をはさんで変数を書きます。

また，〈?〉を出さないようにしたい場合は，〈"〉で囲んだ文字列と変数の間に〈,〉をはさみます。

INPUT文の使い方

INPUT　A

キーボードから入力したデータを，数値として数値変数Aに入れます。

数値データ → 変数A

```
INPUT  A
?123■
```

INPUT　X$

キーボードから入力したデータを，文字列として文字変数X$に入れます。

文字データ → 文字変数XS

```
INPUT  X$
?FORESIGHT■
```

INPUT"アナタ　ノ　ナマエ"；Y$

まず，"アナタ　ノ　ナマエ？"と表示します。それからキーボードから入力したデータを，文字列として文字変数Y$に入れます。

文字データ → 文字変数Y$

```
INPUT"アナタ ノ ナマエ";Y$
?アナタ ノ ナマエ?キムラ カナエ—
```

このプログラムでは，INPUT 文は210行で使ってます。
LIST 命令で，210行を見てみると一。

やっぱり。ちゃんと，使ってました。
これで，あなたの名前，が変数 Y$に…。この変数 Y$の内容はプログラムのもう少しあとの方で使っています。

PRINT は，出力です

今度は，

PRINT文

について，説明します。
PRINT*文は，**定数や変数の値を画面に表示する**ための命令です。
とても，大切な命令。BASIC の思ってることを教えてもらう大切な命令，です。

PRINT	画面表示

働き

画面に計算結果などの情報を表示します

文法

PRINT {定数／変数／式} [{；／，} {定数／変数／式} ……]

説明

PRINTは，画面上に定数，数値変数の値，文字変数の値（文字列），および式の値（演算結果）を出力します。

● PRINT（プリント）
働き 画面に計算結果などのデータを表示します。
文法 PRINT〔定数または変数または式〕…定数，変数，式→画面に表示させるデータ
説明 PRINT は，画面上に定数，数値変数の値，文字変数の値（文字列），および式の値（演算結果）を出力します。
PRINT によって出力される値は〈；〉か〈，〉で区切ることによって複数個続けて書くことができます。
区切り記号として〈；〉を使うと，文字をすぐ後に続けて書くことができます。
ただし，数値の前の符号の部分と後ろには必ず〈 〉を１つずつとります。
区切り記号として〈，〉を使うと，８文字単位で画面上に続けて出力されます。

PRINTによって出力される値は〈;〉か〈,〉で区切ることによって複数個続けて書くことができます。この時の〈;〉や〈,〉を区切り記号（セパレータ）といいます。
区切り記号として〈;〉を使うと，文字をすぐ後に続けて書くことができます。ただし，数値の前の符号の部分と後ろには必ず〈 〉が1つずつとられます。
区切り記号として〈,〉を使うと，8文字単位で画面上に続けて出力します。画面の表示領域は4つのブロックに分割されていて，出力情報は必ず各ブロックの頭から表示されます。
PRINT文の最後に〈;〉をつけると，次のPRINT文の内容が接続して表示されます。
PRINTを単独で使うと，1行改行が行われます。

このプログラムではPRINT文を，**わたしの名前(木村香奈枝)**とみなさんの名前を表示するのに，使っています。
ただ，PRINT文は，画面のどこに表示するか，までは決められないんです。これじゃあ，思っちゃいます。ちゃんと，ハートのまん中に，わたしの名前と…。
なので。
表示する位置を決めるように，

LOCATE文

を使ってみました。
この「LOCATE文」は**カーソルの位置を決める命令**だそうです。この命令で表示したい位置を決めて。それから，PRINT文で表示すると，思った通りの場所に表示できます。

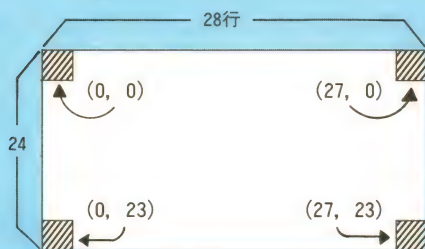
LOCATE	カーソルを設定

働き

カーソルを指定位置へ移動します。

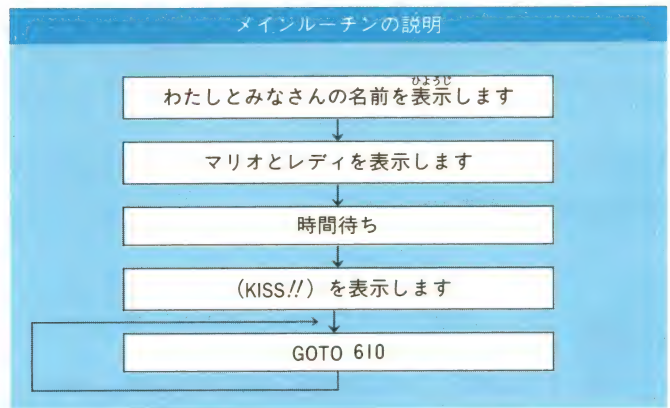
文法

LOCATE　x, y

x→水平方向のカーソル位置（0〜27）
y→垂直方向のカーソル位置（0〜23）

このプログラムでは，次のようにして，LOCATE 文と PRINT 文を組み合わせて使いました。

```
500 '* メインフ゜ロク゛ラム  *
520 LOCATE 10,4:PRINT Y$
530 LOCATE 10,6:PRINT"キムラ カナエ"
```

ENDで，終わります

このプログラムの実行を止めたい時，STOP キーを押します。
いつまで待っても，このプログラムは終わりません。終わり，ないです。
これは**無限ループ**に入っちゃってるから…。メインルーチンのフローチャートを見ると，よく分ります。

こんなふうになってたらず〜っと「GOTO　610」を繰り返してばかり。いつまでたっても，終りません。
だから STOP キーを，押します。
でも。
STOP キーを押さなくても，プログラムの方からすすんで実行をストップする方法が，あるんですよ。
それは

END命令

です。
「END 命令」は**プログラムの実行をやめる命令**だそうです。
BASIC は，この命令に出合うと，実行するのをやめてしまい

ます。べつに，めんどうだからじゃあないんです。

END	実行終了

働き

プログラムの実行の終了を宣言します。

文法

END

説明

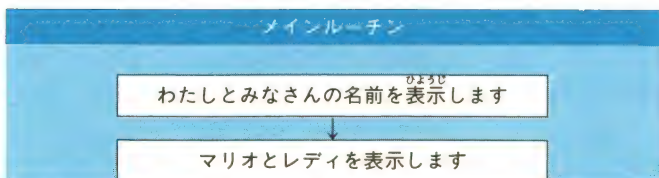
ENDはプログラムの実行を終了します。

このステートメントを実行するとプログラムを終了し，コマンド待ちの状態になります。ENDはプログラムを終了させたいところに，どこにでも置くことができます。

プログラム最終行のEND命令は省略できます。

このEND命令を　610行の

```
610 GOTO 610
```

の代わりに入ります。

こうすると。

610行の「END 命令」に，BASIC は出会います。

すると，BASIC は「お…，これで終わりだな」と思って，実行を終了(しゅうりょう)します。

代入文は＝なんです

代入文，説明をします。

でも。

わたし　この「代入文」を，プログラムで使うの忘(わす)れちゃってました。**ごめんなさい…！**　なかなか，わたしの思う通りに書けません。

それでも，大切な命令，です。

ちゃんと説明したい，と思います。

なので，このプログラムでは使ってませんが，

代入文

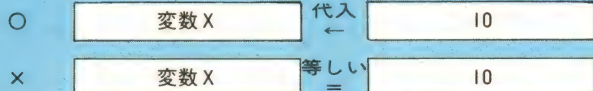
の説明をしたいと思います。

代入文は，**いろいろ計算した結果(けっか)を変数に入れる** ための命令です。

例えば　**X=10** という代入文は **変数Xに10を代入してる**ということ。　**変数Xが10と等しい，ということじゃない** です。

本当は　X=10　よりも　**X←10**　と書く方が良いそうです。でも〈←〉なんて記号ないもん，しかたないです。

代入文　**代入**

働き

変数に式の値を代入します。

文法

変数＝式*

説明

代入は，〈＝〉で結ばれた右辺の左の値を，左辺の変数値に代入します。

右辺の式には単なる定数や変数も含まれ，左辺には必ず変数のみが書かれます。右辺の変数は，その前（直前とはかぎりません）の行で定義されている必要があります。もし，定義されていなければ，変数は数値変数のときに0，文字変数のときに〈　〉とみなされます。

また〈＝〉で結ばれる両辺の型は，一致していなければなりません（文字変数と数値変数の混用はできません)。

例

○　　X＝A　Y＝A＊X＋10　X$＝"A"

×　　X＝A$　X＝"&H"△$

この代入文も使って，変数Xの値に１を加えたい時は，

```
X = X + 1
```

とします。

それから，変数Aに変数B$の文字数を入れたい時は **LEN(n)関数***を使って

```
A =LEN (B$)
```

● 式

式とは，定数や変数を演算子で結合した一般的な数式力，単に文字や数値，あるいは変数だけのものをいいます。

(例)

"BASIC"

314

10＋3/5

A＋B/C－D

RND (A)

● LEN

働き 文字列の文字数を与えます

文法 LEN (文字列)

説明 文字列に含まれるすべての文字数が，この関数の値になります。文字数は0～31で，文字列がスルストリング（何もない文字列）の時は0になります。空白やコントロールコードなどの画面表示されない文字も1文字として数えられます。

● MOD

剰余の計算はMODによって行われます。結果は，整数の割り算の余りです。

(例)

13　MOD　4＝1

 3　MOD　2＝1

● ASC

働き 文字コードを数値に変換します。

文法 ASC (文字列)

文字列→数値に変換する文字が左にある文字列

説明 文字列の最初の1文字のキャラクタコードが，この関数の値になります。キャラクタコードの値は0～255です。文字列は，式でも変数でもかまいません。

文字列がヌルストリング（文字列が無い）の場合，この関数の値は0になります。

● CHR$

働　き 数値をキャラクタコードとみなして，対応する文字に変換します。

文　法 CHR$（x）

x→文字に変換する数式（0〜255）

説　明 数値をキャラクタコードとする文字が与えられます。数値1つにつき1文字が得られます。プリント表示される文字，記号は32〜255です。

● RND（ランド）

働　き 引数未満の乱数を発生します。

文　法 RND（x）

x→数式（1〜32767）

説　明 発生した乱数が，この関数の値になります。この乱数は「引数（x）－1」までの乱数になっています。RND（1）は常に0です。

というふうにします。

第4章

かならず面白いミニ・ゲーム・サンプル集

なんと、この章では「ミニゲーム」を紹介しながら、プログラミング・テクニックを説明してしまいます。うーん、ちょっとサービスのしすぎ、と思ったりもします。でも、せっかくなので、サービスしちゃいます！　4章の担当は、木村香奈枝です。なお、背のおっきめの子が木村香奈枝、背のちっちゃめの子が斉藤千秋、です。あまり関係のないこと、書いてしまいました。どうも、ごめんなさいです。

ミニ・ゲーム集です

1章，2章，3章…。

そしてこの**「4章」**です。

1章からここまで，**ず〜っと**読んでくれた人。

本当に，嬉しいです。わたし，どんなふうに「お礼」を言えばいいのか，分からないです。ほんの少し，みなさんのお役に立てたと思えることが，**しあわせ**です。

なんとなく**ペラペラ**って読んでくれた人。

それでも，とても嬉しい，です。わたしの書いた「字」が，机でコツコツ書いた字が，みなさんの目に触れただけで，**満足。**

ぜんぜん読まなかった人。

ごめんなさい! わたし，もう少し文章が上手だったら，と思います。もっと上手になるように，一生懸命勉強します…。

それで，この4章では

いろんなミニ・ゲーム

を作ってみたいと，思います。

憧れちゃいます

一つめは

READ文、DATA文

を使った，とても簡単なゲームを作ってみました。

命令の名前	意　味
READ	データを読み込む命令です
DATA	データを用意しておく命令です
RESTORE	読み込む場所を決める命令です

```
LIST
500 '* メインルーチン *
510 CLS:INPUT"アナタ ノ ナマエ";I$
540 A=0
550 FOR I=1 TO LEN(I$)
560 J$=MID$(I$,I,1)
570 IF J$=" " THEN A=I
580 NEXT
600 A$=LEFT$(I$,A-1)
610 B$=RIGHT$(I$,LEN(I$)-A)
620
630 PRINT"------------------------------------
640 RESTORE
650 J=RND(20)+1
660 FOR I=1 TO J
670 READ I$
680 NEXT
690
700 PRINT
710 PRINT"*アナタ ノ ミドルネーム ハ"
720 PRINT B$;" ";I$;" ";A$
730 PRINT"ト イウノハ ドウデショウ?"
750
800 PRINT
810 PRINT"[illegible](Y/N)"
820 I$=INKEY$
830 IF I$="" THEN 820
850 IF I$="N" THEN 630
860 PRINT"SEE YOU AGAIN!!"
870 END
900 '* データ *
910 DATA"[illegible]","[illegible]","[illegible]
","[illegible]","ロタロー","ハ"クハツタ"[illegible]
920 DATA"[illegible]","キタロー","ネズ"ミ
[illegible]","[illegible]","スナカケハ"ハ"ー
930 DATA"MILK","LEMON","アップ°ル
ハ"イ",+chr$(34)+[illegible]
940 DATA"PC-8001","ZX","MZ-8
OK","APPLE II","M5","X1ターホ"
OK.
```

500	注釈(ちゅうしゃく)です
	どうして最初で〝メインルーチン〟なんて，REM文があるというと。初め，もっとキレイな画面にしようかな，と思ってました。なので，100〜400行には，キレイな画面を作るプログラム，作るつもりでしたの。

↓

510-610	名字と名前に分けます
	みなさんの名前を入力してもらって。その名前を，名字と名前に分けます。A$←名字，B$←名前。

↓

630-680	ミドルネームを選びます
	たくさんのミドルネームから，一つだけ選びます。

↓

800-870	名前を表示(ひょうじ)します
	名前に，ミドルネームをはさんで，表示します。 気に入ってもらえなかったら，もう一度，やりなおし。

↓

900-940	データ
	ミドルネームのデータです。

このプログラムは　みなさんの名前に，ミドルネームを勝手につけてしまいます。

ミドルネームって外国の人がよくつけてます。例えば，エドガ・アラン・ポー。この「アラン」が，ミドルネームだそうです。

なかなか，素敵(すてき)。憧(あこが)れちゃいます。ミドルネームって…。

うーん　わたしなら…。

木村　〝セイラ〟香奈枝(かなえ)。木村〝ミミィ〟香奈枝。ああ。ぜんぜん似合(にあ)わないんです，もう。やっぱり　外国の人じゃなきゃ，

むりみたい，です。
　そこで，

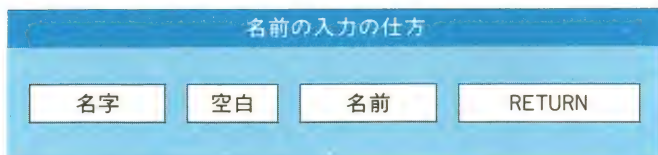
ミドルネームをつけるプログラム

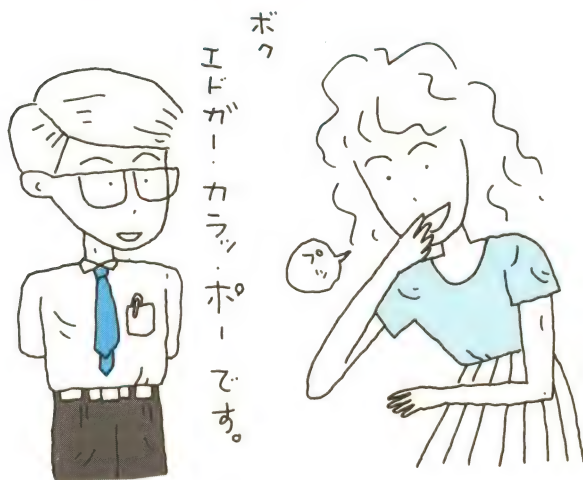
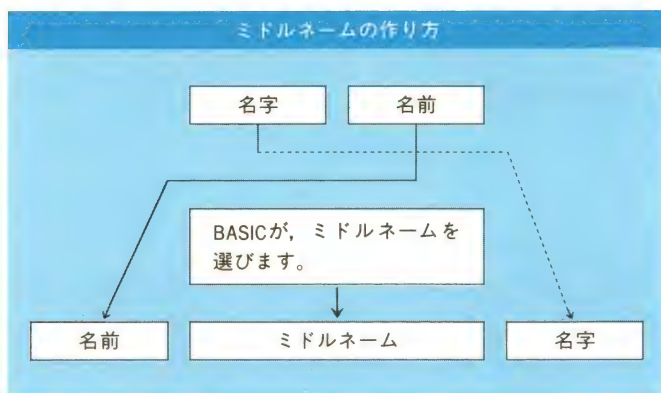
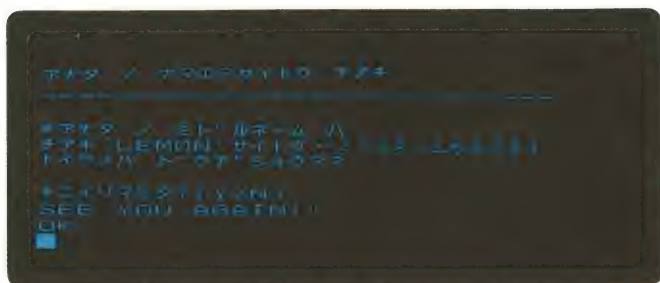
を作ってみました。

木村“ポンタ”香奈枝です

　まず，RUN 命令で実行します。

```
RUN [RETURN]
```

　そうすると，「あなたの名前」を尋ねてきます。みなさんの名前を，入力してあげてくださいね。
　ただ，名字と名前の間には，必ず空白〈 〉を入れます。そうしないと，BASIC は，どれが名字で，どれが名前なのか，分らなくなってしまいます。これじゃあ，BASIC が，かわいそうです。

　このように入力すると，あとは自動です。すべて，BASIC がやってくれます。
　BASIC は，ミドルネームを付けて画面に表示します。もしも，

気に入らなかったら〈N〉を入力します。

データは，DATA です

このプログラムでは，**READ 文，DATA 文，RESTORE 文**を，使っています。

DATA 文は，READ 文で **読むデータを用意しておく命令。** READ 文は，DATA 文の中の **データを読み出す命令** です。

それから，RESTORE 文は，READ 文が **読む場所を決める命令** だそうです。

READ	データを読む

働　き

DATA で用意されたデータを READ 文の変数に入力します

文　法

READ　変数 [，変数，変数…]

変数→ DATA 文のデータを入れる変数

説　明

データの入力には INPUT 文の他に，もう１つ READ 文があります。INPUT 文は，プログラム実行時にデータを入力しますが，実行のたびに決まったデータを用いる場合など，不便なので，このような場合には READ 文を使うようにします。READ 文は，DATA 文といつも対にしていっしょに使うステートメントで，READ のあとに変数を書き，DATA のあとにはそれに対応する定数データを書きます。READ 文の変数と DATA 文の定数データは１対１に対応しかつ双方とも同じ型でなくてはなりません。

DATA 文はプログラム中のどこにあってもかまわず，１つの READ 文で２つ以上の DATA 文からデータを読み込んだり，いくつかの READ 文の１つの DATA 文を共用して読み込むこともできます。いずれの場合も DATA は行番号の若い順に，データ並びの先頭から読み込まれます。

DATA 文のデータの数が READ 文の変数より多い場合，次に来る READ 文によって引き続き読み込まれ，READ 文がなければ残りのデータは無視されます。逆に，DATA 文のデータの数が不足している場合，「ON　ERROR」のエラーメッセージが出ます。

DATA	データを用意

働　き

READ で読み込むデータを用意します

文　法

DATA 定数 [, 定数, 定数, …]
定数→数値定数 (−32768〜+32767), 文字定数 (文字列)

説　明

DATA 文は, READ 文で読み込むデータを用意するステートメントです。DATA 文によって用意できるデータは数値定数および文字定数で, 1 行につき255文字分入力可能です。
何かを実行するステートメントではないので, プログラム中のどこにでも, いくつでも置けます。

RESTORE	読み込む場所指定

働　き

READ 文で読む DATA 文を指定します

文　法

BASIC インタプリタは, DATA を指すポインタを持っており, READ 文を実行すると, そのプログラムの先頭から DATA 文を探して行き, 最初に出てきた DATA にポインタをセットします。RESTORE を実行すると, DATA 用のポインタが指定された行番号の行にセットされます。RESTORE のみのときは, ポインタがプログラムの先頭にセットされます。

READ, DATA の使い方
100 RESTORE 310
200 READ A, A$
300 DATA 1, AB
310 DATA 2, BC
変数 A
2
変数 A$
BC

タイプ練習プログラム

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	L 32	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62
1	L 42	K 72	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52
2	L 42	K 62												
3	L 42	K 62												
4	L 42	K 62												
5	L 42	K 62												
6	L 42	K 62												
7	L 42	K 62												
8	L 42	K 62												
9	L 42	K 62												
10	L 42	K 62												
11	L 42	K 62												
12	L 42	K 62												
13	L 42	K 62												
13	L 42	K 62												
14	L 42	K 62												
15	L 42	K 62												
16	L 42	K 62												
17	L 42	K 62												
18	L 42	K 62												
19	L 42	L 12	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52
20	L 52	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62

14	15	16	17	18	19	20	21	22	23	24	25	26	27	
L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 72	0
K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	L 02	L 42	1
												K 62	L 42	2
												K 62	L 42	3
												K 62	L 42	4
												K 62	L 42	5
												K 62	L 42	6
												K 62	L 42	7
												K 62	L 42	8
												K 62	L 42	9
												K 62	L 42	10
												K 62	L 42	11
												K 62	L 42	12
												K 62	L 42	13
												K 62	L 42	14
												K 62	L 42	15
												K 62	L 42	16
												K 62	L 42	17
												K 62	L 42	18
K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	K 52	L 22	L 42	19
L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	L 62	M 02	20

ペンペンとその仲間(なかま)たち

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	H 72		H 72		H 72		H 72		H 72		H 72		H 72	
1			I 60	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30
2	H 72		J 20											
3			J 20											
4	H 72		J 20			F 70								
5			J 20											
6	H 72		J 20											
7			J 20								F 70			
8	H 72		J 20											
9			J 20											
10	H 72		J 20											
11			J 20											
12	H 72		J 20											
13			J 20											
14	H 72		J 20				I 43	I 13	I 23					
15			J 20				I 43	I 43	I 43					
16	H 72		J 24		F 70						F 70			
17			J 24											
18	H 72		J 20											
19			J 20											
20	H 72		J 00	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30

14	15	16	17	18	19	20	21	22	23	24	25	26	27	
H 72		H 72		H 72		H 72		H 72		H 72		H 72		0
J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	I 70		H 72	1
											J 20			2
							F 70				J 20		H 72	3
											J 20			4
											J 20		H 72	5
	I 43	I 13	I 23								I 20			6
	I 43	I 43	I 43								J 20		H 72	7
											J 20			8
							F 70				J 20		H 72	9
											J 20			10
		F 70									J 20		H 72	11
											J 20			12
											J 20		H 72	13
						I 03	I 13	I 23			J 20			14
						I 43	I 43	I 43			J 20		H 72	15
	F 70										J 20			16
											J 20		H 72	17
											J 20			18
											J 20		H 72	19
J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 30	J 10			20

ハートで、すきして!

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0										M 72	M 72	M 72		
1									M 72	M 72	M 72	M 72	M 72	
2								M 72	M 72	M 72	M 72	M 72	M 72	M 72
3								M 72	M 72	M 72	M 72	M 72	M 72	M 72
4								M 72	M 72	M 72	M 72	M 72	M 72	M 72
5								M 72	M 72	M 72	M 72	M 72	M 72	M 72
6									M 72	M 72	M 72	M 72	M 72	M 72
7										M 72	M 72	M 72	M 72	M 72
8											M 72	M 72	M 72	M 72
9												M 72	M 72	M 72
10													M 72	M 72
11														M 72
12														
13														
14														
15														
16														
17														
18														
19	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00
20	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50

14	15	16	17	18	19	20	21	22	23	24	25	26	27	
	M 72	M 72	M 72											0
M 72	M 72	M 72	M 72	M 72										1
M 72	M 72	M 72	M 72	M 72	M 72									2
M 72	M 72	M 72	M 72	M 72	M 72									3
M 72	M 72	M 72	M 72	M 72	M 72									4
M 72	M 72	M 72	M 72	M 72	M 72									5
M 72	M 72	M 72	M 72	M 72										6
M 72	M 72	M 72	M 72											7
M 72	M 72	M 72												8
M 72	M 72													9
M 72														10
														11
														12
														13
														14
														15
														16
														17
														18
F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	F 00	19
K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	K 50	20

あとがき

●千秋のあとがき

あー，えーと，ごめんなさい！

この本で，ぜんぜん文章を書かなかったんだもん，ひどい。ほとんど，香奈枝ちゃんが1人で書いてくれました。本当は，もっともっと，たくさん書くはずだったのに。

でも，書けなかったんです。

わたしらしい文章って，どんなふうに書くのか，わからなかったんです。もしも，もう一度チャンスがあったら，ちゃんと書きます。指きりげんまん，約束しますー。

スタッフの方々，遅れたのは，わたしの責任，です。ごめんなさい。

この本を読んでご不満のある人，わたしの責任です。ごめんなさい。

でも，楽しかった，です。

●香奈枝のあとがき

わたしたちに，こんなお話が舞い込んだのは，去年の夏の終わり頃。残暑が，あつくって，あつくって…。

おぼえています。14才，でした。

それから，一年近くたった，なんて。

今では，もう，15才，背たけは，ますます，伸びてきちゃって…。

本当は，4月頃には，みなさんの目の前に…，のはずでした。お正月から，はじめてたんですよ。

遅れちゃった理由，文章とプログラミング。

どっちも，思うようには，できなかったんです。あ、今でも，やっぱり，だめ，ですけど…。

陸奥A子先生，西村玲子先生，高橋はるみ先生，何回も何回も，文章を写して，勉強させていただきました。

スタッフの方，そして，この文章を見てくださっているみなさん，ありがとうございました！

読んでくださった皆様方。もしも，よければ，感想や，気がついたこと，お手紙ください。

もっと良い本を出したいと思います。

(一生懸命)　詳細・明解パソコンシリーズ

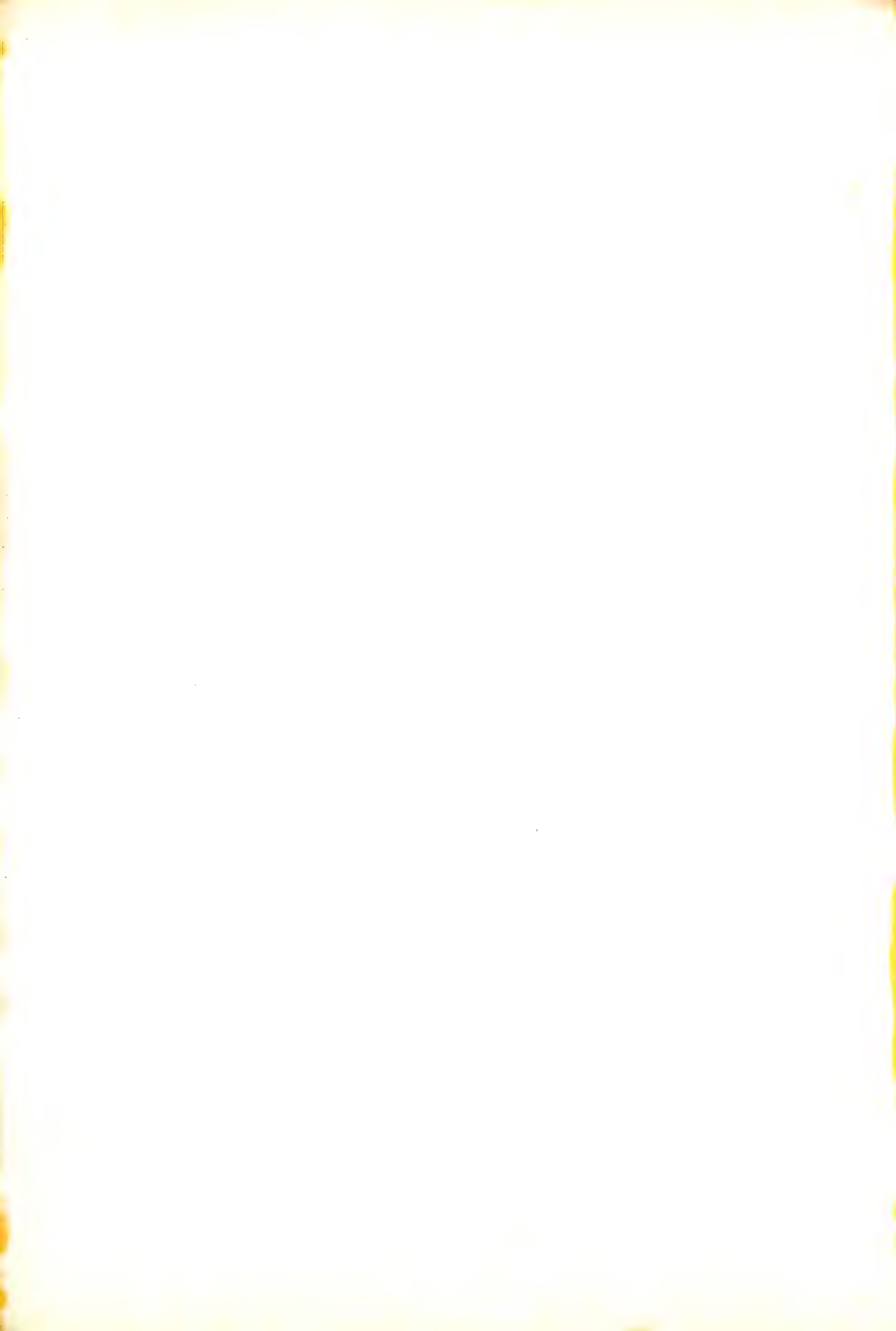
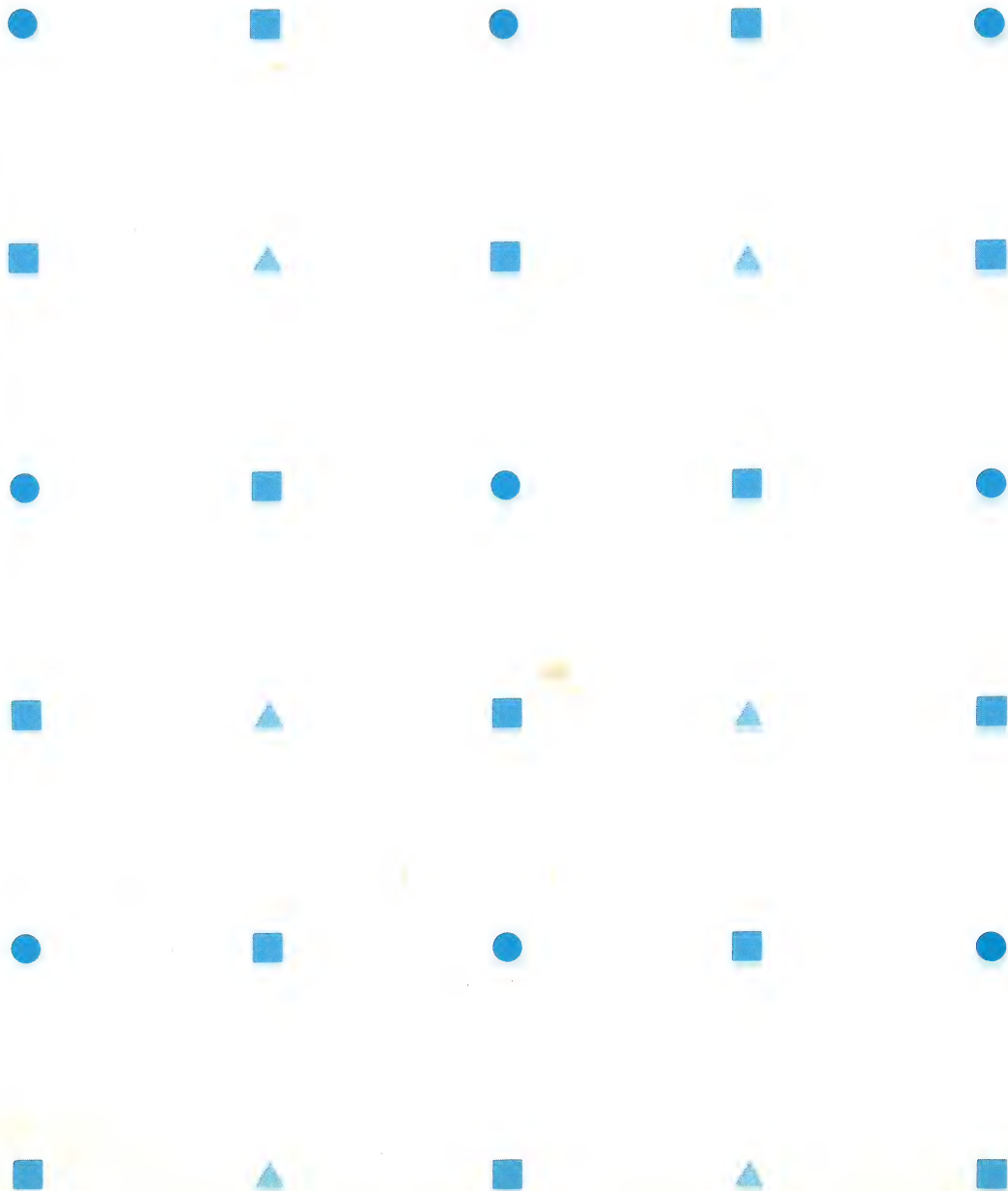
ファミリーコンピュータ

ゲーム＋（プラス）ベーシック入門

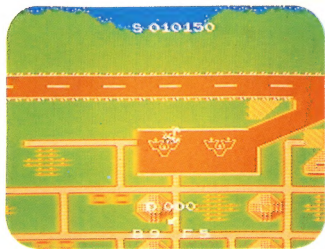
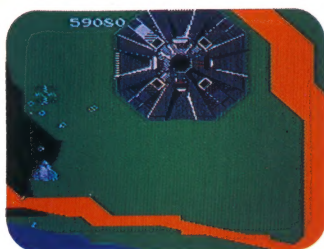
著者────木村香奈枝・斉藤千秋
企画協力───小牧自行
発行者────田村正隆

発行所────[株式会社]ナツメ社
東京都千代田区神田神保町1－52
電話〈03〉291－1257　振替＝東京3－58661
印　刷───ラン印刷社
製　本───三修紙工

製作協力───本田しのぶ・スタジオMADOKA・高橋晴美
技術協力───沼田裕・川村清・フォーサイト企画部スタッフ

ISBN4-8163-0516-5　●定価はカバーに表示してあります。

一生懸命（いっしょうけんめい） 詳細・明解パソコンシリーズ
ファミリーコンピュータ™
ゲーム＋（プラス）ベーシック入門

著　　者	斉藤千秋・木村香奈枝
企画協力	小牧自行
発 行 者	田村正隆
発 行 所	ナツメ社 東京都千代田区神保町1-52 電話 東京03(291)1257 振替口座 東京3-58661
定　　価	980円

落丁・乱丁本はお取り替え致します。

第1章　カートリッジ・ゲーム　ぜったい攻略法

- ゼビウス
- ギャラガ
- F1 レース
- バンゲリング ベイ
- けっきょく南極大冒険
- イー アル カンフー
- V3.0について

ティー・タイム　ふんわり、フワッフワ、背のびしたいナ

第2章　ファミコン・マイコンかんぜん基礎知識

- ファミコンの中身は…
- CPUとRAMとROM
- BASICは、オムレツです
- プログラムは、行の集まりです

第3章　よくわかる最高水準BASICと入門講座

- 入力は、難しくないです
- すごい、タッチメソッド
- 遊びながら、入力上手になれます
- ダイレクトとプログラム
- 変数、です！
- 大切な命令だから

第4章　かならず面白いミニ・ゲーム・サンプル集

- ミニ・ゲーム集です

定価 980円

ISBN4-8163-0516-5 C2054 ¥980E

一生懸命（いっしょうけんめい）
ファミリーコンピュータTM
ゲーム＋（プラス）ベーシック入門
ナツメ社

一生懸命（いっしょうけんめい）
ファミリーコンピュータ™
ゲーム＋（プラス）ベーシック入門
ナツメ社

詳細・明解
パソコンシリーズ

ファミリーコンピュータ™
ゲーム＋(プラス)ベーシック入門

木村香奈枝・斉藤 千秋 著

●本書をお読みの方へ

●本書は**ファミリーコンピュータ**をよりよく利用するための入門書です。特に**ファミリーベーシック**をお持ちで、BASICを理解したいと思われている方には最適です。

●カートリッジゲームの特選攻略法、正当的キーボード入門法『タッチメソッド』など、ファミコンを活用するうえでのすべてを解説してあります。

●本書の特長は「やさしく詳しく」説明してあることです。やさしく、詳しい。この一見反対のようなことを、本文では必要なことをやさしく、注釈文では本文では説明できなかった解説を詳しく説明。注釈文については技術スタッフが担当しています。

●このように「やさしく詳しい」本書は低年令の方から大人の方までじゅうぶんにご満足いただけることでしょう。